Design and Performance Evaluation of 1 Giga Hertz Wideband Digital Receiver

Kiranraj George
Wright State University

Follow this and additional works at: https://corescholar.libraries.wright.edu/etd_all

Part of the Engineering Commons

Repository Citation
https://corescholar.libraries.wright.edu/etd_all/135

This Dissertation is brought to you for free and open access by the Theses and Dissertations at CORE Scholar. It has been accepted for inclusion in Browse all Theses and Dissertations by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.
DESIGN AND PERFORMANCE

EVALUATION OF 1 GIGA HERTZ WIDEBAND DIGITAL RECEIVER

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

By

KIRANRAJ GEORGE
B.S., Bharathiar University, India, 1999
M.S., Wright State University, 2001

2007
Wright State University
WRIGHT STATE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

June 28, 2007


____________________________________
Chien-In Henry Chen, Ph.D.
Dissertation Director

____________________________________
Fred D. Garber, Ph.D.
Chair, Department of Electrical Engineering

____________________________________
Joseph F. Thomas, Jr., Ph.D.
Dean, School of Graduate Studies

Committee on Final Examination

____________________________________
Chien-In Henry Chen, Ph.D.

____________________________________
Marian Kazimierczuk, Ph.D.

____________________________________
Raymond Siferd, Ph.D.

____________________________________
Jack S.N. Jean, Ph.D.

____________________________________
Wen-Ben Jone, Ph.D.
ABSTRACT


The lack of a priori knowledge about the waveform of interest, the multitudes of signals the receiver might receive, and the noise energy that occupies the same portion of the frequency spectrum as the signal makes the design of a modern wideband receiver very challenging. Especially, the receiver must be able to detect a weak signal in the presence of a strong one, which requires a high two-signal instantaneous dynamic range (IDR). To fulfill this requirement, the receiver must detect genuine weak signals and avoid the detection of strong signals’ sidelobes and noise and spurs generated from the receiver system. The other major trend in modern receiver signals is the shift towards wider bandwidths. Analog wideband receiver designs can provide accommodation of the technology-stressing bandwidths, but come up to a cost of reduced flexibility. Digital approaches, alternatively, provide flexibility in receiver signal processing, but they are limited by analog-to-digital converter resolution and power consumption.

In this dissertation, design and performance evaluation of a 1-GHz signal bandwidth digital receiver, which uses a Kaiser Window function and a compensation technique, is presented. The Kaiser Window reduces the spectral leakage by eliminating the discontinuities at the time window edges and the compensation uncovers the weak signal for extension of the two-signal IDR of the receiver. An exhaustive study of configuration
of ADC, FFT, window function, and compensation for a maximum achievable two-signal IDR of the receiver is conducted. It is shown that using a 4-bit ADC and a 256-point FFT of 12-point kernel function, a maximum two-signal IDR of 9 dB is obtained. The IDR is extended to 14 dB by using the Kaiser window and to 18 dB by using the compensation. Furthermore, using a 4-bit ADC and an ideal 256-point FFT, a maximum two-signal IDR of 11 dB is obtained. The IDR is extended to 17 dB by using the Kaiser window and to 22 dB by using the compensation. A combination of both Kaiser window and compensation techniques extends the two-signal IDR of the receiver to 24 dB by using a 12-point kernel function FFT and 29 dB by using an ideal FFT. A novel hardware implementation of the Kaiser window, the compensation technique, and the receiver design is presented.
# Contents

## I. Introduction

1.1 Wideband Digital Receivers .................................................................................................................. 2
1.2 Problem Statement .................................................................................................................................. 5
1.3 Dissertation Scope and Methodology ......................................................................................................... 6
1.4 Summary .................................................................................................................................................. 7

## II. Literature Review

2.1 Introduction ............................................................................................................................................... 8
2.2 Digital Wideband Receiver .......................................................................................................................... 8
2.3 Data Windowing ......................................................................................................................................... 12

## III. Digital Receiver Design

3.1 Introduction ................................................................................................................................................. 14
3.2 Proposed Digital Receiver ........................................................................................................................... 16
   1) Signal Sampler and Formatting System ................................................................................................. 16
   2) Super-Resolution and Frequency Measurement System .................................................................... 19
      a) FFT Design ..................................................................................................................................... 19
      b) Compensation ................................................................................................................................. 20
      c) Compensation in Three Frequency Case ....................................................................................... 23
      d) Frequency Measurement .............................................................................................................. 24
3.3 Experimental Results .................................................................................................................................. 29
3.4 Summary .................................................................................................................................................. 35

## IV. Extension of Two-Signal Instantaneous Dynamic Range (IDR) using Kaiser Window and Compensation Method

4.1 Introduction .................................................................................................................................................. 36
4.2 Data windowing ........................................................................................................................................... 39
4.3 Receiver-On-a-Chip II (ROC II) .................................................................................................................... 53
4.4 Kaiser window and compensation ................................................................................................................ 56
   1) Effect of Normalization on Compensation ......................................................................................... 56
   2) Effectiveness of Kaiser Window Function and compensation ....................................................... 57
   3) Hardware Design and Implementation ............................................................................................... 59
      a) Kaiser Window Function ................................................................................................................ 59
      b) Super-Resolution Block ................................................................................................................ 60
4.5 Timing for Super-Resolution Block ............................................................................................................ 77
4.6 Simulation Results ....................................................................................................................................... 78
4.7 Chip Implementation .................................................................................................................................... 88
4.8 Summary .................................................................................................................................................. 89

## V. Multiple Signal Detection and Measurement Using a Configurable Wideband Digital Receiver

5.1 Introduction .................................................................................................................................................. 90
5.2 Digital Receivers ....................................................................................................................................... 91
List of Figures

3.1 Two areas of ROC ................................................................. 18
3.2 12-point kernel function for 256 point FFT ................................. 21
3.3 Architecture of the super-resolution and frequency measurement .............................. 22
3.4 Architecture of the super-resolution for three signal case ...................................... 24
3.5 Examples of signal detection and false alarm .................................. 25
3.6 Flowchart for single signal detection .............................................. 27
3.7 Flowchart for two signals detection ............................................... 28
3.8 Two signal frequencies of 435 and 923 MHz ................................ 30
3.9 Second signal detection probability with and without compensation ....................... 31
3.10 Second signal false alarm with and without compensation ......................... 31

4.1(a) 10 Hz sine waveform with integer number of cycles and its Fourier transform ........ 39
4.1(b) 10 Hz sine waveform with non-integer number of cycles and its Fourier transform ................................................................. 40
4.1(c) A transient 10 Hz sine waveform and its Fourier transform ......................... 40
4.2 Time windowing .............................................................................. 42
4.3 Magnitude spectrum of rectangular window ........................................ 42
4.4 Spectra of windowed single and double sinusoids .................................... 45
4.5 Rectangular and non-rectangular windows in time and frequency domains ........ 46
4.6 Time and frequency domain plot of Hann window .................................. 47
4.7 Time and frequency domain plot of Hamming window ............................ 48
4.8 Time and frequency domain plot of Blackman window ......................... 48
4.9 Spectra comparison of Fixed non-rectangular windows ......................... 49
4.10 Time and frequency domain plot of Chebshev windows ....................... 51
4.11 Time and frequency domain plot of Kaiser window ................................ 52
4.12 Spectra comparison of adjustable non-rectangular windows ................. 53
4.13 A time domain data series divided into 256-sample FFT slices ................. 53
4.14 Two areas of ROC II ........................................................................ 54
4.15 Architecture for the Kaiser Window block ....................................... 55
4.16 Effect of normalization on compensation ........................................... 57
4.17 The effectiveness of Kaiser Window and compensation ......................... 58
4.18 Example of “Kaiser coefficient LUT 102” ......................................... 60
4.19 Super-Resolution block ...................................................................... 61
4.20 Normalization sub-block ...................................................................... 62
4.21 Multiplication by splitting operands .................................................. 63
4.22 General flow of LUT multiplier ......................................................... 63
4.23 LUT construction for 4 x 4 Multiplication ........................................ 64
4.24 Pipelined 8 x 8 multiplier .................................................................. 66
4.25 16x16 LUT multiplier ........................................................................ 67
4.26 Bit format for 1/Xmax ........................................................................ 71
4.27 Inverse LUT construction ................................................................. 71
4.28 Example of normalization ..................................................................... 72
4.29 Block diagram for Find “n” block ....................................................... 74
4.30 The effect of compensation with different row selection .................................................. 75
4.31 Compensation LUT block ............................................................................................... ..77
4.32 Timing for Super-Resolution block .................................................................................. 78

5.1 False alarm due to noise .................................................................................................... 92
5.2 Block diagram of digital receiver (ROC II) ...................................................................... 92
5.3 Block diagram of the configurable receiver ...................................................................... 93
5.4 Flowchart of multiple signal detection ............................................................................. 94
5.5 Example of five signal detection ...................................................................................... 96
5.6 Illustration of IDR of five signal scheme ......................................................................... 97
5.7 Maximum attainable IDR for two, three, four and five signal cases ................................. 98
5.8 Signal detection, miss, and false alarm for the five signal case ........................................ 99
5.9 Frequency error distribution for detected signals ......................................................... 100

6.1 Functional blocks of the digital receiver ......................................................................... 105
6.2 Conventional radix-r butterfly architecture ..................................................................... 106
6.3 Receiver performance of conventional FFT with bit truncation ................................... 106
6.4 Conventional FFT spectrum with bit truncation ............................................................ 107
6.5 Receiver performance of 12 pt kernel FFT with bit truncation ....................................... 109
6.6 12 pt kernel FFT spectrum with bit truncation .............................................................. 109
6.7 Radix-2 butterfly architecture using LUT .............................................................. 110
6.8 Radix-2 LUT construction ............................................................................................ 112
6.9 Frequency detection error of the second signal .............................................................. 115
6.10 Frequency detection error of the second signal for the optimum bit truncation scheme .................................................................................. 117
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Comparison of typical IFM, Monobit receiver and ROC</td>
<td>17</td>
</tr>
<tr>
<td>3.2</td>
<td>Results of two simultaneous signals</td>
<td>32</td>
</tr>
<tr>
<td>3.3</td>
<td>Results of three simultaneous signals</td>
<td>33</td>
</tr>
<tr>
<td>4.1</td>
<td>Comparison of properties of different window functions</td>
<td>50</td>
</tr>
<tr>
<td>4.2</td>
<td>Performance comparison of the proposed LUT Multiplier</td>
<td>70</td>
</tr>
<tr>
<td>4.3</td>
<td>The Sensitivity of Row Estimation in Compensation matrix for 10,000 Simulation Runs</td>
<td>75</td>
</tr>
<tr>
<td>4.4</td>
<td>Two-signal IDR Results</td>
<td>79</td>
</tr>
<tr>
<td>4.5</td>
<td>Transistor Count for each Subsystem</td>
<td>80</td>
</tr>
<tr>
<td>4.6</td>
<td>ROC II Statistics</td>
<td>80</td>
</tr>
<tr>
<td>4.7</td>
<td>Maximum achievable two signal IDR with consideration of amplifier output noise and bit truncation ROC II Statistics</td>
<td>86</td>
</tr>
<tr>
<td>6.1</td>
<td>Optimal bit truncation for each stage of conventional FFT computation</td>
<td>108</td>
</tr>
<tr>
<td>6.2</td>
<td>Gate count and timing for two radix-2 butterflies in different stages of 256-point FFT</td>
<td>113</td>
</tr>
<tr>
<td>6.3</td>
<td>Gate count and delay estimates before and after LUT Configuration for 1024, 512, 256, and 128-point FFT</td>
<td>114</td>
</tr>
<tr>
<td>6.4</td>
<td>Signal detection, miss and false alarm of the second signal using LUT FFT</td>
<td>116</td>
</tr>
</tbody>
</table>
Acknowledgements

This Doctoral dissertation would not have been possible without the encouragement and goodwill of many people. My most sincere thanks to my dissertation advisor, Dr. Henry Chen, for his invaluable support and guidance throughout my entire dissertation. He has truly helped me lay a sound foundation for my career not only as a competent researcher but also as an effective educator. He has indeed helped me tremendously to further my career goals, and to him I remain humbly indebted. I would also like to sincerely thank my committee members, Dr. Marian Kazimierczuk, Dr. Raymond Siferd, Dr. Wen-Ben Jone, and Dr. Jack Jean for evaluating my dissertation and for being on my Final Dissertation Examination Committee. I extend my acknowledgements to Sheila Hollenbaugh and the Department of Electrical Engineering for providing me all the necessary technical support required for the successful completion of this dissertation. I would also like to thank Dayton Area Graduate Studies Institute (DAGSI) for the Competitive Full Scholarship and the Engineering Ph. D. Program for their financial support.

I am most grateful to my parents for their constant love and unstinting faith in me. Above all, I thank God for being my steadfast pillar of strength and hope.
Dedicated to

My best friend and partner in life, Sheeba.
I. INTRODUCTION

Today’s very deep submicron IC technology enables high-performance analog and digital applications to be integrated on a single piece of silicon. Currently, there is a growing interest in the VLSI implementation of wideband digital receivers because of the advancement in analog-to-digital converters (ADC) and high-speed digital signal processing (DSP). These advances in digital electronics technology have lead to continuous pressure on analog receivers in terms of competitiveness in performance, flexibility and programmability. However, it will be some time before the digital receivers will compete with analog systems in terms of instantaneous bandwidth. An analog IFM receiver can cover an instantaneous bandwidth of 16 GHz (from 2 to 18 GHz) and a practical approach with today’s technology to build a digital receiver with comparable instantaneous bandwidth would be through channelization.

The use of digital channelization in comparison with the analog approach allows improving the imbalances between filters, which is one of the fundamental problems in analog receivers. However, broadband digital channelized receivers, mainly based on discrete Fourier transform (DFT) related processing is less suitable for real-time applications as the speed of the ADC and the computational complexity of the DSP following it become bottleneck in the design. In an attempt to improve the real-time
operation, high-speed ADCs and signal processing algorithms with reduced computational complexity has to be employed. If such a receiver can be realized, its performance may be superior in comparison to a conventional analog approach because of three major differences:

1) More information can be extracted in the digital approach. The ADCs can replace the crystal detector which used to be placed before the DSP in the past and thus keep valuable information like fine frequency and phase information that would be lost otherwise.

2) The digitized data can be stored for long periods of time unlike conventional method of storage like magnetic tapes.

3) Signal processing for digitized signal is much advanced and flexible compared to the analog signal processing.

1.1 Wideband Digital Receivers

Desirable characteristics of digital receiver include wideband input bandwidth with fine frequency resolution, high sensitivity and instantaneous dynamic range (IDR), simultaneous signal detection, and full real-time operation. A wide input bandwidth is desirable to reduce the search time in the spectrum of interest and thus improve the response time. Also if the signal is phase shift keying (PSK) or Frequency shift keying (FSK), it is desirable to have a high enough instantaneous bandwidth to cover a spectrum range containing the entire signal. Otherwise, part of the signal information may not be properly collected. The maximum number of simultaneous signals a digital receiver is
required to process is often considered as four. This means that if four pulses arrive at the
receiver at the same instant of time, the receiver should obtain information on all the
signals. High IDR is a highly desirable property which enables the detection of weak
signal detection in presence of a strong signal.

Table 1.1: Comparison of typical IFM and Monobit receiver.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>IFM Receiver</th>
<th>Monobit Receiver [8]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluation Method</td>
<td>Measured</td>
<td>Simulated</td>
</tr>
<tr>
<td>RF Front-End</td>
<td>Limiting Amp</td>
<td>Limiting Amp</td>
</tr>
<tr>
<td>Sampling Rate (GHz)</td>
<td>NA</td>
<td>2.5</td>
</tr>
<tr>
<td>No. of ADC Bits</td>
<td>NA</td>
<td>2</td>
</tr>
<tr>
<td>FFT</td>
<td>NA</td>
<td>256-point FFT of 4-point kernel function</td>
</tr>
<tr>
<td>Bandwidth (GHz)</td>
<td>2 – 16</td>
<td>1</td>
</tr>
<tr>
<td>Sensitivity (dBm)</td>
<td>Medium</td>
<td>-70</td>
</tr>
<tr>
<td>Number of Signals</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Single Signal Dynamic Range (dB)</td>
<td>70</td>
<td>75</td>
</tr>
<tr>
<td>2 Signal Spurfree DR (dB)</td>
<td>NA</td>
<td>75</td>
</tr>
<tr>
<td>2 Signal IDR (dB)</td>
<td>NA</td>
<td>5</td>
</tr>
<tr>
<td>Channel Bandwidth (MHz)</td>
<td>2000 –16000</td>
<td>10</td>
</tr>
<tr>
<td>Frequency Measurement (MHz)</td>
<td>1</td>
<td>+/- 5</td>
</tr>
<tr>
<td>Frequency Resolution (MHz)</td>
<td>NA</td>
<td>10</td>
</tr>
<tr>
<td>Time Resolution (ns)</td>
<td>NA</td>
<td>100</td>
</tr>
<tr>
<td>Amplitude</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Minimum Pulse Width (ns)</td>
<td>100</td>
<td>200</td>
</tr>
</tbody>
</table>
The Monobit receiver design [8-11] was an attempt to increase processing speed by avoiding general multiplication in the DFT. This was achieved by using single-bit digital representation of the input signal and using a Monobit representation of the kernel of the DFT. The receiver which includes a 2-b ADC and a 256 point FFT implemented using a 4-point kernel function could simultaneously detect and process two signals in 1 GHz bandwidth (125 MHz ~ 1.125 GHz) and achieve a two signal instantaneous dynamic range of 5 dB. Monobit receiver is unlike the conventional instantaneous frequency measurement (IFM) receiver which only detects one signal. Table 1.1 compares the performance of typical IFM receiver and Monobit receiver.

In the Monobit receiver design, the input signal is down converted into intermediate frequency (IF), which is then sampled using high-speed ADCs followed by digital signal processor to extract the data. The rate at which the signal is sampled decides the degree of preservation of the information content and thus in turn the dynamic range and resolution of the receiver. The choice of the step size for the quantizer accounts for the noise level of the input signal. Otherwise, the noise present in the input signal will trigger the ADC and thus produce an erroneous output. But if the step size is too large, the quantizer will miss some weak signals.

Detection of the target signal is established by checking if the receiver’s output exceeds a preset threshold level. Thus if the output signal exceeds the threshold, the target is declared as present. The threshold level is very critical since it affects the detection probability of the receiver and thus directly its performance. If the threshold is too high, then the receiver misses the weak signals and if it’s too low the receiver picks
up sidelobes or noise spurs instead of signals. This is a situation similar to the choice of the step size for the quantizer.

1.2 Problem statement

Modern receiver’s function is to intercept the RF signals in order to identify and locate its source. Because of the complex nature of the electromagnetic emissions present, the receiver is required to rapidly search through a large frequency range with maximum sensitivity. Since the time that can be spent to examine a particular segment of frequency spectrum is limited, practical receivers are only capable of scanning through a small segment of the total range of frequency spectrum at any instant of time.

The lack of *a priori* knowledge of the waveform of interest is a major limiting factor specific to the receiver. This is due to the multitudes of signals the receiver might receive and thus isolation of the waveform of interest becomes a critical issue. Also the noise energy, which occupies the same portion of the frequency spectrum as the signal, complicates the matter even further. This in combination with other sources of spurious signals will reduce the maximum achievable dynamic range of the wideband receiver. The threshold could be increased to reduce the chances of spurs being picked up as signals, but then the sensitivity of the receiver is compromised.

Nyquist sampling criterion limits the achievable input bandwidth. In order to cover 1-GHz bandwidth for real data, the ADC must operate at least 2 GHz. The disparity in the performance in terms of maximum operating speeds of the ADC and the digital signal
processor is also another impediment in the receiver design implementation. Current operating speeds of the digital hardware do not yet match the speed of the state-of-art ADC.

The parameter encoder which retrieves useful information like carrier frequency is an important component of a digital receiver. Even though there are several spectral estimation schemes proposed for parameter encoding, many of them are usually computationally intensive for real-time application.

1.3 Dissertation scope and Methodology

Monobit receiver was an attempt to realize a channelized digital receiver by using an efficient scheme to reduce the computational complexity. The major advantage of the Monobit receiver is its simplicity and efficient hardware. The price for simplicity is performance trade-off in terms of achievable two signal dynamic range and the maximum number of simultaneous signals the receiver can process. Innovative techniques and algorithms for high precision spectral estimation and dynamic range extension are explored and presented to reduce the above mentioned deficiencies of the Monobit receiver. This is achieved by improving the infrastructure of the Monobit design, based on the performance goal.

Methodologies to convert a design from software into hardware would be explored after its performance evaluation in the software environment is satisfied. This would involve iterative software-hardware co-simulation to come up with the optimum
hardware which can perform real-time without compromising the efficiency of the algorithm.

The literature review in chapter II is a brief overview of the applicable literature and relevant current research in the areas of wideband digital receivers and data windowing. In chapter III, the design of a 1 GHz wideband Receiver-On-a-Chip which uses compensation technique to improve two signal IDR is presented. Chapter IV discusses extension of IDR using data windowing and compensation and its hardware implementation. Chapter V introduces a multiple signal detection scheme and its implementation. A configurable and expandable FFT processor is presented in chapter VI.

1.4 Summary

The research goal is to conduct design and performance evaluation of a 2.5 giga-sample per second (GSPS) digital receiver, which: 1) extends a maximum attainable two signal IDR, and 2) increases the maximum number of simultaneous signals the receiver can process for a given frequency resolution. This research is accomplished by exploring innovative and efficient algorithms coupled with software and hardware simulations and analysis.
II. LITERATURE REVIEW

2.1 Introduction

Literature pertaining to Digital Wideband receivers and Data windowing are discussed below. This literature review is intended to provide direction to the sources of information to understand the broad context of the research.

2.2 Digital Wideband Receiver

Modern digital radar receivers are required to search for weak signals in presence of very strong signals, and external and internal noise [12]. Unlike the narrow-band communication receivers, which usually only receive one signal at a time and designed for a known signal, modern radar receivers does not know the modulation or the carrier frequency of the incoming signals. This makes the implementation of these receivers a challenge.

In the past, most receivers were designed using analog technology in which the RF signal is converted into a video signal, and in this process, information like carrier frequency is lost. But in a digital receiver, the signal is down converted to a lower frequency and digitized. This process retains all the information and thus the digital
receiver can produce better results than an analog one [12]. One other attractive feature of a digital receiver is its ability to perform complex signal processing on the input signal that is difficult to do with conventional analog counterpart [13].

The ADC is a critical subsystem of a digital receiver which sets the receiver’s maximum bandwidth and instantaneous dynamic range, especially for receivers which use direct IF sampling [14]. The ADC error sources such as quantization noise and spurious components are introduced, and their effects on dynamic range of digital receiver are analyzed in [15]. Due to the vigorous research in ADCs the operating speed and the number of resolution bits are increasing at a speed that the DSPs cannot catch up with. But, the DSPs in the Digital Receivers of the future must process data at a rate commensurate with the ADC for a faster, efficient and a real-time operation [13][16]. One possible temporary remedy until then would be to multiplex the ADC outputs coming at a high data rate and feed the DSP chip at a slow data rate [8]. Another approach would be to use multirate digital processing [16-19] or combine several narrowband receivers to achieve a wide bandwidth [16]. Design of the data processing block based on a filter bank is proposed in [20]. A digital filter bank design based on the fast Fourier transform (FFT) with a numerical example is presented in [12]. A 3 GHz ADC with 8-bits and a 256 point FFT is used in the design. The achievable frequency resolution is about 93.75 MHz, which is too coarse for signal sorting since the desirable frequency resolution is approximately less than 10 MHZ for wideband communications.

Digital spectral analysis is an integral part of a digital receiver that involves determination of power spectrum of the signal. The spectral analysis methods are based on the observation that if a continuous time signal is bandlimited, the spectral
characteristics of its discrete-time equivalent should provide a good estimate of its spectral properties [21]. FFT is a popular spectral analysis method widely used in digital receivers. Several high-resolution spectrum estimation algorithms were discussed in the literature [22-25]. Sampling with weighted integration is employed in [26] to increase the dynamic range of the digital receiver. Many of them can provide higher frequency resolution than FFT, especially on simultaneous signals. But they suffer from a major drawback of being computationally complex and thus cannot be used for real-time applications [16].

Various techniques have been proposed in the past to reduce the computational complexity of data processing in digital receivers, especially in spectral analysis [27][28]. The original motivation of the Monobit receiver design [8-11] was to increase processing speed by avoiding general multiplication and keep only adders and shifters, in the discrete Fourier transform (DFT). Several analysis and evaluations on performance and limitations of the Monobit receiver has been presented in [27][29]. An improvement to the Monobit receiver was proposed in [2] to extend the two signal instantaneous dynamic range to 18 dB with the second signal false alarm less than 1%. The receiver design included a 4-b ADC [30], a 256 point FFT implemented using a 12-point kernel function. A super-resolution block implementing the compensation technique was presented to improve the instantaneous dynamic range of the receiver by subtracting the detected peak amplitude frequency response and its spurs from the actual response and thereby exposing the second weak signal [31][32]. By using this technique a pre-calculated FFT output of the highest amplitude frequency ($f_i$) was pre-stored in a look-up table (LUT). The super-resolution algorithm is then applied to estimate the $f_i$ frequency with a higher
resolution which then determines and retrieves the correct frequency response from the compensation LUT for subtraction from the actual response.

Multiple signal detection is one of the highly desirable characteristic of wideband receivers, and often a modern receiver is required to detect up to four simultaneous signals [12]. Some of the bottlenecks of multiple signal detection are noisy signal environment, signal interferences, and computational complexity involved in a real-time implementation. In the past several multiple signal detection schemes have been proposed. Multiple signal detection using optimum filters in presence of noise was presented in [33]. The scheme compensates for the match filters incapability to process more than two signals. It addresses the problem of deterioration in the receiver performance with the increase in the number of signals to detect. The detection problem for multiple signals embedded in noisy environment is treated as a multiple hypothesis test based on log-likelihood ratios [34]. A scheme based on the sequentially rejective Bonferroni test together with the nonlinear weighted least squares approach for estimation of the signal parameters under hypotheses and alternative is presented in [35]. Multiple signal detection using the atomic decomposition and the expectation maximization (EM) algorithms was proposed in [36]. The proposed approach addresses the need of advanced signal processing algorithms for counterbalancing increasing complexity of the communication and radar signals.
2.3 Data Windowing

Real-world data is rarely constant over time and often data needs to be analyzed in a succession of short time slices. The basic Fourier integrals only work properly for periodic signals. A signal frequency which is not periodic in the Fourier analysis is still perceived by the analysis as cyclical. Consequently, the analysis typically sees step disjoints in the waveform with a periodicity equal to the analysis period. This disjoints lead to a glitches and spurious responses in single-frequency analyses. One possible solution is to window the source-data. This removes the end-effect glitches, and introduces a well-behaved pseudo-periodicity.

Window functions have played a vital role in DSP and found extensive applications in signal analysis and spectral estimation [37-40]. The commonly used non-rectangular fixed windows, such as Hamming, Hann and Blackman-Harris, are variations on the basic idea of smoothing with a raised cosine sequence.

The rectangular window clearly has the narrowest main lobe, and thus for a given length, it yields the sharpest transitions at the discontinuities. By tapering the windows smoothly to zero, as with non-rectangular windows, the side lobes are greatly reduced in amplitude; however the price paid is a much wider mainlobe. The mainlobe width of the windowed signal reduces with the length of the window [41]. Several windows have been proposed to offer various degrees of trade-off between $\Delta f$ and peak sidelobe level. This trade-off can be quantified by seeking a window function that is maximally concentrated around $\omega = 0$ in the frequency domain. This issue was considered in depth in a series of classic papers [42]. Kaiser found a near optimal window which is good approximation to
a class of optimal windows called prolate spheroidal wave sequences [43]. The Kaiser
Window function has two parameters: the length and a shape parameter $\beta$. By varying the
length and $\beta$, the shape can be adjusted to trade side-lobe amplitude for main-lobe width
which has a great impact on wideband receiver performance.
III. DIGITAL RECEIVER-ON-A-CHIP

3.1 Introduction

A modern receiver’s function is to intercept the RF signals in order to identify and locate its source. Because of the complex nature of the electromagnetic emissions present, the receiver is required to rapidly search through a large frequency range with maximum sensitivity. Since the time that can be spent to examine a particular segment of frequency spectrum is limited, practical receivers are only capable of scanning through a small segment of the total range of frequency spectrum at any instant of time.

The lack of a priori knowledge about of the waveform of interest is a major limiting factor specific to the receiver. This is due to the multitudes of signals the receiver might receive and thus isolation of the waveform of interest becomes a critical issue. Also the noise energy, which occupies the same portion of the frequency spectrum as the signal, complicates the matter even further.

The weakest signal that can be detected is referred as minimum detectable signal. Detection of the target signal is established by checking if the receiver’s output exceeds a preset threshold level. Thus if the output signal exceeds the threshold, the target is declared as present. The threshold level is very critical since it affects the detection probability of the receiver and thus directly its performance. If the threshold is too high,
then the receiver misses the weak signals and if it’s too low the receiver picks up signal
d sidelobes or noise spurs instead of signals.

The signals are continuous in time and amplitude. The ADC encodes these signals by
sampling them and then quantizing the amplitude. The rate at which the signal is sampled
decides the degree of preservation of the information content and thus in turn the
dynamic range and resolution of the receiver. The choice of the step size for the quantizer
accounts for the quantization noise level of the input signal. Otherwise, the noise present
in the input signal will trigger the ADC and thus produce an erroneous output. But if the
step size is too large, the quantizer will miss some weak signals.

An IFM receiver has a wide instantaneous radio frequency (RF) bandwidth, sometimes
as much as several octaves. The receiver can measure short pulses with high frequency
accuracy (i.e., 1 MHz resolution on 100 nsec pulse). A conventional IFM receiver is
limited to process only one signal. However, if two signals arrive at the receiver
simultaneously, the receiver may generate erroneous information without the operator
knowing. Various techniques have been used to detect the existence of simultaneous
signals or detect the existence of erroneous frequency, but only limited success has been
accomplished.

The original motivation of the monobit receiver design [8-11] was to increase

\[ X(\Delta) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi \Delta n}{N}} \]

processing speed by avoiding general multiplication and keep only adders, in the discrete
Fourier transform (DFT). An analysis and characterization of a monobit receiver was
presented in [27][29]. The DFT can be written as [59][60] where N is the total number of
sampled input points. In this equation the result is obtained from the product of two functions: the input $x(n)$ and the kernel function $(e^{j/\Phi})$ is $e^{-j2\pi kn}$. This objective was realized by replacing the 256 point DFT kernel function with an approximating kernel function of four values (+1, -1, +j, -j), called 4-point kernel function. Because of the nonlinear nature of this approximation, spurious responses are generated that limit the two signal instantaneous dynamic range to the level of 5 db. This section presents a ROC design using compensation technique, which can be used for some special applications or to enhance or supplement other receivers. The ROC has an input bandwidth of 1-GHz and can correctly process two simultaneous signals. The 1-GHz wideband digital receiver is capable of detecting two simultaneous signals in a dense RF environment, providing basic parameter estimation in those signals and providing accurate cueing for higher performing analysis receivers. Table 3.1 compares the performance of typical IFM receiver, monobit receiver and the proposed ROC. This section presents 1) the technical approach to design a digital receiver, 2) the super-resolution technique, and 3) the experimental results of two signal instantaneous dynamic range.

3.2 Proposed Digital Receiver

The design of ROC can be divided into two areas: 1) signal sampler and formatting and 2) super resolution and frequency measurement, as shown in Fig. 3.1.

1) Signal Sampler and Formatting System
Table 3.1: Comparison of typical IFM, Monobit receiver and ROC.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>IFM Receiver</th>
<th>Monobit Receiver</th>
<th>ROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluation Method</td>
<td>Measured</td>
<td>Simulated</td>
<td>Simulated</td>
</tr>
<tr>
<td>RF Front-End</td>
<td>Limiting Amp</td>
<td>Limiting Amp</td>
<td>Linear Amp</td>
</tr>
<tr>
<td>Sampling Rate (GHz)</td>
<td>NA</td>
<td>2.5</td>
<td>2.5</td>
</tr>
<tr>
<td>No. of ADC Bits</td>
<td>NA</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>FFT</td>
<td>NA</td>
<td>256-point FFT of 4-point kernel function</td>
<td>256 point FFT of 12-point kernel function</td>
</tr>
<tr>
<td>Bandwidth (GHz)</td>
<td>2 – 16</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Sensitivity (dBm)</td>
<td>Medium</td>
<td>-70</td>
<td>-70</td>
</tr>
<tr>
<td>Number of Signals</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Single Signal</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dynamic Range (dB)</td>
<td>70</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>2 Signal Spurfree DR (dB)</td>
<td>NA</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>2 Signal Instantaneous DR(dB)</td>
<td>NA</td>
<td>5</td>
<td>18</td>
</tr>
<tr>
<td>Channel Bandwidth (MHz)</td>
<td>2000 –16000</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Frequency Measurement (MHz)</td>
<td>1</td>
<td>+/- 5</td>
<td>+/- 5</td>
</tr>
<tr>
<td>Frequency Resolution (MHz)</td>
<td>NA</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Time Resolution (ns)</td>
<td>NA</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Amplitude</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Minimum Pulse Width (ns)</td>
<td>100</td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>
In order to achieve 1 GHz input bandwidth, the Nyquist sampling frequency required is 2 GHz. To take the finite slope of the input filter into consideration, 2.5 times the input bandwidth is often used. Thus, the ADC should operate at 2.5 GHz. The input signal is first passed to the ADC, which samples the signal every 0.4 ns to produce 4-b amplitude measurements. Each bit is then passed to an associated windowing circuitry, which collects 16 sample serial window of data and outputs the data in parallel to the detection circuit. Thus, the windowing circuitry has two key functions: 1) converts the serial data stream to parallel, and 2) slows down the data rate by factor of 16, i.e., (2.5-GHz sampling rate)/(16-sample window) = 156.25 MHz sampling rate. A clock divider is employed to down rate the data stream. The slowing of the data rate is necessary to accommodate the speed at which the detection circuit can receive data.
2) Super-Resolution and Frequency Measurement System

a) FFT Design

If the points of the kernel function [11] are restricted to coordinates that are integer powers of 2, multiplication can be realized by a simple bit shift operation. For example, the number 3 can be represented by 0011 in a binary system. If 3 is multiplied by 2, the result in 6 which can be represented by 0110 which is a binary shift of one position to the left of the binary number for 3. This design uses 12-point kernel FFT since it shows remarkable improvement in performance compared to 4-point kernel function of monobit receiver. The 12-point kernel function has values of \((2, 2+j, 1+2j, 2j, -1+2j, -2+j, -2, -2-j, -1-2j, -2j, 1-2j, 2-j)\), as shown in Fig. 3.2. The FFT contains 256 points. Sampling at 2.5 GHz, the total time is approximately 100 ns, which can be considered as the minimum pulse width (PW). The FFT block inputs are four 16-b data windows at a rate of 156.25 MHz. The input stage receives and stores each 16-bit data window until 16 windows have been collected, i.e., total data is \((16 \text{ windows}) \times (16 \text{ data samples/window}) = 256 \text{ data samples} (each \text{ sample is four \text{ bit wide})}. Thus, a complete data set is ready every \((16 \text{ windows}) \times (6.4 \text{ ns per window}) = 102.4 \text{ ns and the window sampling circuitry is feeding the FFT every (0.4 nsec ADC sampling rate) x (16 samples) = 6.4 ns.}\)

In order to avoid multiplication in the calculation of FFT the original kernel function \(e^{j\Phi}\) is replace by \(Kernel_{12 \text{ point}} (e^{j\Phi})\). The implementation of 256-point FFT becomes:
$$X(k) = \sum_{n=0}^{N-1} x(n) * Kernel_{12-point}(e^{j\Phi})$$

$$Kernel_{12-point}(e^{j\Phi}) = \begin{array}{|c|c|}
\hline
2 & -0.231824 \leq \Phi < 0.231824 \\
2 + j & 0.231824 \leq \Phi < 0.785375 \\
1 + 2j & 0.785375 \leq \Phi < 1.33897 \\
2j & 1.33897 \leq \Phi < 1.802619 \\
-1 + 2j & 1.802619 \leq \Phi < 2.35617 \\
-2 + j & 2.35617 \leq \Phi < 2.90977 \\
\hline
-2 & 2.90977 \leq \Phi < 3.141593 \\
-2 & \text{or} \\
-2 & -3.141593 \leq \Phi < -2.90977 \\
\hline
-2 - j & -2.90977 \leq \Phi < -2.35617 \\
-1 + 2j & -2.35617 \leq \Phi < -1.802619 \\
-2j & -1.802619 \leq \Phi < -1.33897 \\
\hline
1 - 2j & -1.33897 \leq \Phi < -0.785375 \\
\hline
2 - j & -0.785375 \leq \Phi < -0.231824 \\
\hline
\end{array}$$

(\Phi \text{ is the phase angle in radians})

b) Compensation

The input to the compensation block is 128 set of values (magnitudes) from the FFT block. Fig. 3.3 shows the architecture of the super-resolution and frequency measurement. Each block shows the operation performed and the hardware components
that would be required to carry out the operation. The architecture of Fig. 3.3 comprises of two main flows. Flow I depicts the flow of finding the frequency and magnitude of the highest amplitude signal. Flow II depicts the flow of finding the frequency and magnitude of the second highest amplitude signal.

Compensation is a possible approach that can provide high dynamic range with acceptable frequency resolution. The compensation method [31][32] involves a stored pre-calculated response that closely approximates the highest amplitude frequency \( f_1 \) response which is subtracted from the actual response. Nearly complete sidelobe cancellation can be achieved with the compensation method provided the approximation is good enough. Only a narrow null window needs be used in this case; in fact, if the
approximation is accurate enough, no window is even necessary. For our design, a 10 MHz window on both sides of $f_1$ is selected.

Subtraction of the compensation vector stored in the compensation matrix will in theory eliminate the sidebands and all spurs lying away from the $f_1$ location, thereby “exposing” the weak second signal. It is assumed that all significant sidelobes and outlying spurs are generated by $f_1$ alone and that an accurate enough estimate of $f_1$ should permit the complete elimination of all spurs and sidelobes. The preferred and chosen compensation method uses a super-resolution algorithm for estimation of the $f_1$ frequency to provide a higher resolution of $f_1$ than the usual 10 MHz resolution of the 128 set of
receiver outputs. The higher resolution $f_i$ is then used to determine the correct row of a compensation matrix for subtraction from the actual response. To implement the compensation, the 128 set of FFT outputs has to be normalized (the highest is “1”) and then compared to estimate the amplitude and frequency of the highest amplitude signal ($f_i$). In flow II, the row in the compensation matrix that closely approximates $f_i$ is subtracted from the receiver response for removing the signal $f_i$ as well as its sidelobes and outlying spurs. Since the compensation may not perfectly remove the spurs of $f_i$, a narrow null window is used. The window size has to be carefully chosen. For our simulation, 10 MHz window on both sides of $f_i$ is considered. Ideally the null window need not be used if a perfect compensation takes place. Without compensation these sidelobes and spurs may get picked up as signals by the receiver and thus give false alarms. With a good compensation approximation a nearly complete sidelobe cancellation may be achieved.

After applying the compensation, the next task is to estimate the amplitude and frequency of the “exposed” second highest amplitude signal ($f_2$). Both magnitudes of $f_i$ and $f_2$ are stored for comparison with the threshold level that determines the number of input signals and is based on the computer simulation results.

c) Compensation in Three Signal Case

A preliminary study of the three frequency case was performed. The strong signal frequency, $f_1$, was varied from 125 MHz to 1,125 MHz. The magnitude of the $f_3$ signal is held fixed at 18 db down and the magnitude of the $f_2$ signal varies from 0 db to 18 db
down. Both signal frequencies, \( f_2 \) and \( f_3 \), were also varied from 125 MHz to 1,125 MHz.

Figure 3.4 describes the case where an initial compensation is performed on the strong signal frequency, \( f_1 \), and then a second compensation is performed on the second largest signal frequency, \( f_2 \) or \( f_3 \). In order for the second compensation to work properly, the response, after the first compensation, must be normalized to unity with respect to the magnitude of the response at \( f_2 \) or \( f_3 \) depending on which response is larger. This renormalization is necessary since the compensation matrix is itself normalized to one.

**Fig. 3.4. Architecture of the super-resolution for three signal case**

**d) Frequency Measurement**
Because of the quantized kernel function of Fig 3.2 and the fact that the ADC used in the receiver has only four bits, the system is basically nonlinear. A nonlinear system is difficult to analyze. Thus, in designing the receiver, data collected from a data collection system are evaluated by simulation. In order to determine the number of input signals, thresholds must be incorporated in the design. In our simulation, a threshold of -22 dBm based on computer simulation results is chosen.

Fig. 3.5 gives examples of signal detection and false alarm. In Fig. 3.5, “d1” denotes the event of the first signal being detected; “fa1” and “fa2” denote the events of the first and second signal false alarms. Let “T” denote the threshold level with which the signal amplitudes will be compared. In Fig. 3.5(a), the first signal is detected (d1) but a false alarm (fa2) for the second signal is reported since its amplitude is greater than the threshold and therefore picked up by the receiver. In Fig. 3.5(b), there are false alarms for
the first (fa1) and the second (fa2) signals since both spur’s amplitudes are greater than the threshold. Fig. 3.5(c) is similar to the case of Fig. 3.5(a) except there is no second signal. Fig. 3.5(d) is a case where the spurs have both the first and the second highest amplitude and thereby generate two false alarms, (fa1) and (fa2).

One signal detection flow chart is shown in Fig. 3.6 where the threshold comparison and the flow of signal detection, signal missing, and signal false alarm are depicted. Initially, the signal frequency ($f_1$) and signal amplitude ($m_1$) are estimated. Next, a conditional check ($C_1$) of ($m_1 >$ threshold) is evaluated in which we account for the signal detection. If $C_1$ is true, then a second conditional check ($C_2$) of difference between the actual signal frequency and the estimated frequency is evaluated if it is within the frequency resolution, 10 MHz. This is to verify if the signal is an actual signal or a spur. The first signal is detected (d1) if $C_2$ is true. Otherwise, it is counted as a false alarm (fa1). If $C_1$ is false, then signal is missing.

Two signals detection flow chart is shown in Fig. 3.7 where the threshold comparison and the flow of signal detection, signal missing, and signal false alarm are depicted. If the conditional check of ($m_1 >$ threshold) is false, then both signals are reported missing. Otherwise, once the first signal frequency ($f_i$) is estimated, the difference between the actual signal frequencies and the estimated frequency is checked if it is within the frequency resolution, 10 MHz. The first signal is detected (d1) if the check is true. Otherwise, it is counted as a false alarm (fa1).

After applying compensation by subtracting the compensation vector from the first signal, the sidelobes and spurs are removed from the 128 set of FFT outputs. The next is
to find the second signal and there are two conditional checks to verify if the found signal is an actual signal or a spurious signal.

Fig. 3.6. Flowchart for single signal detection
In the first conditional check of \((m_2 > \text{threshold})\), we account for the signal detection if the second highest amplitude \((m_2)\) is greater than the threshold. If the conditional check
is true, we’ll check if the second highest amplitude \( (m2) \) signal is an actual signal (the difference between the estimated frequency \( (f_2) \) and the actual signal frequencies is within 10 MHz). If the second conditional check is true, then the second signal is detected \( (d_2) \). Otherwise, the second signal is a false alarm \( (fa_2) \). If the first conditional check is false \( (m2 < \text{threshold}) \), the second signal is reported missing.

### 3.3 Experimental Results

The two signals generated for the simulation have random frequencies (between 125 MHz and 1125 MHz) and random phases. A pre-calculated response that closely approximates the highest amplitude frequency response is stored in a compensation matrix of size, 1024 x 128. The sharp sensitivity of the \( f_1 \) response to small changes in \( f_1 \) is observed particularly in the sideband structure. Since the changes in \( f_1 \) are smaller than the resolution interval, it is clear that some attempt must be made to obtain a reasonably accurate estimate of the \( f_1 \) lying between two bin frequencies in order to gain access a higher resolution compensation matrix. Based on computer simulation results, we used a 1024 x 128 compensation matrix.

Fig. 3.8 illustrates an example of two signals detection which underlines the effectiveness of the super-resolution technique. Fig. 3.8 gives a receiver response for \( f_1 = 923 \) MHz and \( f_2 = 435 \) MHz and illustrates the presence of two sidelobes of \( f_1 \) (923 MHz) with and without compensation. With the compensation the two strong sidelobes and spurs are nearly completely cancelled. The second signal is exposed and picked up by the
receiver. However, without compensation, the receiver would have picked up the spurious signal as the second signal.

Fig. 3.8. Two signal frequencies of 435 and 923 MHz

Table 3.2 shows the simulation of two signals with different instantaneous dynamic ranges, after compensation. The signal detection, signal missing, and false alarm are reported in the subsequent columns. The minimum frequency separation is 10 MHz (one channel width) and the maximum amplitude separation is set up to 18 dB. It is our observation that if instantaneous dynamic range is greater than 18 dB, the receiver has a false alarm of greater then 1%. At each of instantaneous dynamic range, 10,000
Fig. 3.9. Second signal detection probability with and without compensation

Fig. 3.10. Second signal false alarm with and without compensation
Table 3.2: Results of two simultaneous signals

<table>
<thead>
<tr>
<th>Magnitude of 2nd signal vs. 1st signal (dB)</th>
<th>1st signal detection (%)</th>
<th>2nd signal detection (%)</th>
<th>2nd signal missing (%)</th>
<th>1st signal false alarm (%)</th>
<th>2nd signal false alarm (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>100</td>
<td>99.35</td>
<td>0.026</td>
<td>0</td>
<td>0.62</td>
</tr>
<tr>
<td>-2</td>
<td>100</td>
<td>99.38</td>
<td>0.04</td>
<td>0</td>
<td>0.58</td>
</tr>
<tr>
<td>-3</td>
<td>100</td>
<td>99.32</td>
<td>0.034</td>
<td>0</td>
<td>0.65</td>
</tr>
<tr>
<td>-4</td>
<td>100</td>
<td>99.19</td>
<td>0.08</td>
<td>0</td>
<td>0.72</td>
</tr>
<tr>
<td>-5</td>
<td>100</td>
<td>99.21</td>
<td>0.47</td>
<td>0</td>
<td>0.74</td>
</tr>
<tr>
<td>-6</td>
<td>100</td>
<td>99.28</td>
<td>0.16</td>
<td>0</td>
<td>0.56</td>
</tr>
<tr>
<td>-7</td>
<td>100</td>
<td>99.23</td>
<td>0.21</td>
<td>0</td>
<td>0.56</td>
</tr>
<tr>
<td>-8</td>
<td>100</td>
<td>99.22</td>
<td>0.21</td>
<td>0</td>
<td>0.57</td>
</tr>
<tr>
<td>-9</td>
<td>100</td>
<td>98.98</td>
<td>0.43</td>
<td>0</td>
<td>0.58</td>
</tr>
<tr>
<td>-10</td>
<td>100</td>
<td>98.98</td>
<td>0.56</td>
<td>0</td>
<td>0.45</td>
</tr>
<tr>
<td>-11</td>
<td>100</td>
<td>98.58</td>
<td>0.88</td>
<td>0</td>
<td>0.54</td>
</tr>
<tr>
<td>-12</td>
<td>100</td>
<td>98.52</td>
<td>1.07</td>
<td>0</td>
<td>0.41</td>
</tr>
<tr>
<td>-13</td>
<td>100</td>
<td>97.93</td>
<td>1.74</td>
<td>0</td>
<td>0.32</td>
</tr>
<tr>
<td>-14</td>
<td>100</td>
<td>97.09</td>
<td>2.62</td>
<td>0</td>
<td>0.29</td>
</tr>
<tr>
<td>-15</td>
<td>100</td>
<td>95.92</td>
<td>3.54</td>
<td>0</td>
<td>0.54</td>
</tr>
<tr>
<td>-16</td>
<td>100</td>
<td>94.54</td>
<td>5.04</td>
<td>0</td>
<td>0.42</td>
</tr>
<tr>
<td>-17</td>
<td>100</td>
<td>89.72</td>
<td>9.62</td>
<td>0</td>
<td>0.66</td>
</tr>
<tr>
<td>-18</td>
<td>100</td>
<td>80.88</td>
<td>18.12</td>
<td>0</td>
<td>0.91</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>100</strong></td>
<td><strong>96.96</strong></td>
<td><strong>2.49</strong></td>
<td><strong>0</strong></td>
<td><strong>0.56</strong></td>
</tr>
</tbody>
</table>
Table 3.3: Results of three simultaneous signals

<table>
<thead>
<tr>
<th>Magnitude of 3rd signal vs. 1st signal (dB)</th>
<th>1st signal detection (%)</th>
<th>2nd signal detection (%)</th>
<th>3rd signal detection (%)</th>
<th>2nd and 3rd signal missing (%)</th>
<th>1st signal false alarm (%)</th>
<th>2nd signal false alarm (%)</th>
<th>3rd signal false alarm (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>100</td>
<td>100</td>
<td>98.29</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.71</td>
</tr>
<tr>
<td>-2</td>
<td>100</td>
<td>100</td>
<td>98.18</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.82</td>
</tr>
<tr>
<td>-3</td>
<td>100</td>
<td>100</td>
<td>98.14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.86</td>
</tr>
<tr>
<td>-4</td>
<td>100</td>
<td>100</td>
<td>98.02</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.98</td>
</tr>
<tr>
<td>-5</td>
<td>100</td>
<td>100</td>
<td>97.98</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.02</td>
</tr>
<tr>
<td>-6</td>
<td>100</td>
<td>100</td>
<td>98.07</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.93</td>
</tr>
<tr>
<td>-7</td>
<td>100</td>
<td>99.99</td>
<td>98.09</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>1.9</td>
</tr>
<tr>
<td>-8</td>
<td>100</td>
<td>99.96</td>
<td>98.01</td>
<td>0.02</td>
<td>0</td>
<td>0.02</td>
<td>1.97</td>
</tr>
<tr>
<td>-9</td>
<td>100</td>
<td>99.96</td>
<td>97.64</td>
<td>0.02</td>
<td>0</td>
<td>0.02</td>
<td>2.34</td>
</tr>
<tr>
<td>-10</td>
<td>100</td>
<td>99.90</td>
<td>97.43</td>
<td>0.03</td>
<td>0</td>
<td>0.07</td>
<td>2.54</td>
</tr>
<tr>
<td>-11</td>
<td>100</td>
<td>99.81</td>
<td>97.15</td>
<td>0.06</td>
<td>0</td>
<td>0.13</td>
<td>2.79</td>
</tr>
<tr>
<td>-12</td>
<td>100</td>
<td>99.61</td>
<td>96.94</td>
<td>0.23</td>
<td>0</td>
<td>0.16</td>
<td>2.83</td>
</tr>
<tr>
<td>-13</td>
<td>100</td>
<td>99.56</td>
<td>96.28</td>
<td>0.23</td>
<td>0</td>
<td>0.21</td>
<td>3.49</td>
</tr>
<tr>
<td>-14</td>
<td>100</td>
<td>99.34</td>
<td>95.28</td>
<td>0.42</td>
<td>0</td>
<td>0.24</td>
<td>4.3</td>
</tr>
<tr>
<td>-15</td>
<td>100</td>
<td>99.10</td>
<td>94.13</td>
<td>0.64</td>
<td>0</td>
<td>0.26</td>
<td>5.23</td>
</tr>
<tr>
<td>-16</td>
<td>100</td>
<td>98.56</td>
<td>92.24</td>
<td>1.15</td>
<td>0</td>
<td>0.29</td>
<td>6.61</td>
</tr>
<tr>
<td>-17</td>
<td>100</td>
<td>97.55</td>
<td>89.02</td>
<td>1.99</td>
<td>0</td>
<td>0.46</td>
<td>8.99</td>
</tr>
<tr>
<td>-18</td>
<td>100</td>
<td>95.91</td>
<td>84.89</td>
<td>3.66</td>
<td>0</td>
<td>0.43</td>
<td>11.45</td>
</tr>
<tr>
<td>Average</td>
<td>100</td>
<td>99.40</td>
<td>95.86</td>
<td>0.47</td>
<td>0</td>
<td>0.12</td>
<td>3.65</td>
</tr>
</tbody>
</table>
simulation runs were taken and each run regenerates two signals for simulation. Only for very few cases, the receiver reads in a spurious signal rather than the actual signal. The average performance of the receiver is: 100% detection of the first signal, 96.96% detection of the second signal, 2.49% of the second signal missing, 0% of the first signal false alarm, and 0.56% of the second signal false alarm. As seen from the table, the highest false alarm is 0.91% when instantaneous dynamic range is 18 dB which is an improvement of 13 dB while compared with the monobit receiver.

Fig. 3.9 and 3.10 depict the second signal detection and the second signal false alarm with and without compensation. Both are plotted against the two signals dynamic range of the receiver. It can be seen that as the instantaneous dynamic range of the receiver is increased, in general the second signal detection probability goes down and the false alarm goes up. It has been shown in Fig. 3.9 the second signal detection probability without compensation drops to 18% at 18 dB of two signal dynamic range. However, the detection probability with compensation is dramatically increased to 81%. Fig 3.10 has shown the second signal false alarm with compensation is less than 1%. However, the false alarm without compensation rises to 82%.

Table 3.3 shows the simulation of three signals with different instantaneous dynamic ranges after compensation. The strong signal frequency, f_1, was varied from 125 MHz to 1,125 MHz. The magnitude of the f_3 signal is held fixed at 18 db down and the magnitude of the f_2 signal varies from 0 db to 18 db down. Both signal frequencies, f_2 and f_3, were also varied from 125 MHz to 1,125 MHz. The signal detection, signal missing, and false alarm are reported in the subsequent columns. At each of instantaneous dynamic range, 10,000 simulation runs were taken and each run regenerates three signals.
The average performance of the receiver is: 100% detection of the first signal, 99.40% detection of the second signal, 95.86% detection of the third signal, 0.47% of the second and third signal missing, 0% of the first signal false alarm, 0.12% of the second signal false alarm, and 3.65% of the third signal false alarm. As seen from the table, the third signal false alarm is greater than 1% for all cases and is less than 3% when instantaneous dynamic range is less than or equal to 12 dB. The number increases when instantaneous dynamic range increases. The results of three-frequency case, although preliminary, are quite encouraging.

### 3.4 Summary

The two major deficiencies of receiver design are limited instantaneous dynamic range and the two signal detection capability. The purpose of the design is to have higher instantaneous dynamic range and the capacity of processing two simultaneous signals with reasonable hardware implementation. Possible approaches to eliminate the two deficiencies are to increase the bit number of the ADC and change the kernel function of the receiver. Increasing the bit number of ADC and the number of bits in the kernel function will require more hardware, which affects the minimum hardware and processing requirements of the digital receiver.

A new compensation algorithm using higher order approximating kernel functions was proposed, analyzed and simulated. This algorithm has the potential of extending the two signal dynamic range to 18 db for an input signal quantized to 4 bits.
IV. EXTENSION OF TWO-SIGNAL INSTANTANEOUS DYNAMIC RANGE (IDR) USING KAISER WINDOW AND COMPENSATION METHOD

4.1 Introduction

The lack of a priori knowledge about the waveform of interest, the multitudes of signals the receiver might receive and the noise energy which occupies the same portion of the frequency spectrum as the signal makes the design of a modern wideband digital receiver very challenging [16][44]. The dynamic range of microwave receivers is ultimately limited by the signal sidelobes and spurs that could ultimately become false alarms.

A practical approach to build a wideband digital receiver is through channelization. The Monobit digital wideband receiver was one such approach first presented in [8] which used a fast Fourier transform (FFT) to perform channelization. The complexity of a 256-point fast Fourier transform (FFT) was reduced by eliminating multipliers and thereby improving the performance of the receiver in terms of processing speed. The Monobit receiver including a 2-b ADC and a 256-point FFT implemented using a 4-point kernel function can simultaneously detect and process two signals in 1 GHz bandwidth (125 MHz – 1.125 GHz) and achieve a two-signal IDR of 5 dB. This receiver is unlike the conventional instantaneous frequency measurement (IFM) receiver only detecting one
signal. Analysis and evaluation of the receiver performance and limitation of the Monobit receiver was presented in [27][29].

A frequency oversampled channelized receiver using hybrid filter banks for ultra-wideband signals were proposed in [45]. Because of its ability to isolate the effects of the narrowband interference, the adaptive performance of the receiver is significantly better than an ideal full-band receiver when narrowband interference is present. Channelization in the wideband receiver of a software defined radio involves the extraction of multiple narrowband channels from a wideband signal using several bandpass filters which is computationally intensive. An arithmetic scheme called as pseudo floating-point (PFP) representation to encode the filter coefficients was proposed in [46]. It was shown that coefficients can be coded using considerably fewer bits than the conventional implementation. A low-power, high-speed implementation of PFP coded filters with minimum hardware was also proposed.

An improvement to the Monobit receiver was proposed in [2] in which the two-signal instantaneous dynamic range was extended to 18 dB with the second signal false alarm less than 1%. The design mainly included a 4-b ADC, a 256 points FFT implemented using a 12-point kernel function and a super-resolution block. The super-resolution block implementing a compensation technique improves the instantaneous dynamic range of the receiver subtracting the estimated peak amplitude frequency response along with its sidelobes and spurs from the actual frequency response, and thereby exposing the weak second signal. By using this technique the pre-calculated FFT outputs of peak amplitude frequencies were stored in a look-up table (LUT). After spectral analysis of the incoming signals, the super-resolution algorithm is applied to estimate the peak signal frequency.
The frequency response of the peak signal is then retrieved from the compensation LUT for subtraction from the actual response.

Real world data is rarely constant over time and often data needs to be analyzed in a succession of short time slices. The basic Fourier integrals only work properly for periodic signals. A signal frequency which is not periodic in the Fourier analysis is still perceived by the analysis as cyclical. Consequently the analysis typically sees step disjoints in the waveform with a periodicity equal to the analysis period. This disjoints lead to glitches and spurious responses in single-frequency analyses. A solution is to window the source-data so as to remove the end-effect glitches and introduce a well-behaved pseudo-periodicity.

The commonly used windows, such as Hamming [37] and Blackman-Harris [38], are variations on the basic idea of smoothing with a raised cosine sequence. They are limited to a fixed stopband ripple that depends on the window shape. Alternatively, adjustable windows, such as the Kaiser Window [39][40], can be used to satisfy alternative design requirements. The Kaiser Window has an advantage of minimizing the sidelobe energy of the window, and a simple implementation. It uses a zeroth-order modified Bessel function [41] of the first kind. It has two parameters: the window length and the shape parameter $\beta$. By varying the window length and $\beta$, the shape of the window can be modified according to the application.

Increasing $\beta$ reduces the amplitude of the sidelobes, but widens the mainlobe. Whereas increasing the window length reduces the width of the mainlobe and increases the amplitude of the sidelobes.
In the next section, a detailed analysis of several fixed and adjustable data windowing functions is given.

4.2 Data Windowing

1) Periodicity of Signals and Fourier analysis

![Fig. 4.1 (a) 10 Hz sine waveform with integer number of cycles and its Fourier transform](image)

The FFT computation assumes that a signal is periodic in each data block, that is, it repeats over and over again and it is identical every time. Fig. 4.1(a) shows a 10 Hz sine waveform and its Fourier transform. In this case, there are an integer number of cycles (10) of the sine wave in the data record. The sine wave is composed of one pure tone indicated by the single discrete peak in the FFT with a height of 1.0 at 10 Hz.
The example in Fig. 4.1(b) shows a sine wave with non-integer number of cycles. Therefore is not periodic and thus the ends of two time frames would have discontinuities.

Fig. 4.1 (b) 10 Hz sine waveform with non-integer number of cycles and its Fourier transform

Fig. 4.1 (c) A transient 10 Hz sine waveform and its Fourier transform
which in turn give rise to an oscillatory behavior in Fourier Transform results called the 
*Gibbs Phenomenon*. This results in spectral leakage when Fourier Transform is 
performed.

A type of signal that satisfies the periodic requirement is a transient signal that starts at 
zero at the beginning of the time window and then rises to some maximum and decays 
again to zero before the end of the time window. The Fig. 4.1 (c) illustrates such a signal. 
When the windowing is applied, the leakage is reduced in the Fourier transform. The 
resulting spectrum is a sharp narrow peak with amplitude of 1.0. Thus a Windowing 
function minimizes the effect of leakage to better represent the frequency spectrum of the 
data.

2) Time-Windowing

Real-world data needs to be analyzed in a succession of short time slices. This process 
is called *time-windowing* [43]. As shown in Fig. 4.2, the time windowed signal can be 
thought of as an infinite signal which is zeroed outside the specified range \(0 \leq n \leq L-1\).

The original sampled spectrum \(X(f)\) and its time-windowed version \(X_L(f)\) are given 
by:

\[
X(f) = \sum_{n=-\infty}^{\infty} x(nT)e^{-2\pi fnT}
\]

\[
X_L(f) = \sum_{n=0}^{L-1} x(nT)e^{-2\pi fnT}
\]

The rectangular window can be mathematically expressed as follows:

\[
w(n) = \begin{cases} 
1, & \text{if } 0 \leq n \leq L-1 \\
0, & \text{otherwise}
\end{cases}
\]
Then, the windowed signal and its Fourier transform can be defined as:

\[
x_L(n) = x(n)w(n) = \begin{cases} 
  x(n), & \text{if } 0 \leq n \leq L-1 \\
  0, & \text{otherwise}
\end{cases},
\]

\[
X_L(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} x(\omega')W(\omega-\omega')d\omega' \quad (1)
\]

where \( W(\omega) \) is the DTFT of the rectangular window \( w(n) \).

Fig. 4.2. Time windowing

Fig. 4.3. Magnitude spectrum of rectangular window
The two major effects of the windowing process are:

1) Reduction in the frequency resolution \((\Delta f = f_s / L, f_s - \text{sampling rate})\) of the computed spectrum.

2) Introduction of spurious high-frequency components into the spectrum caused by the sharp clipping of the signal at the left and right ends of the window.

The magnitude spectrum of the rectangular window is shown in Fig. 4.3. It consists of a mainlobe of height \(L\) and a base width of \(4\pi/L\) centered at \(\omega = 0\) and several sidelobes causing frequency leakage.

For further analysis, let us consider the case of a single analog sinusoid of frequency \(f_1\) and a linear combination of two sinusoids with frequencies \(f_1\) and \(f_2\) separately.

**Case 1: Single Sinusoid**

**Complex Signal**

\[
x(t) = e^{2\pi i f_1 t}, \quad -\infty < t < \infty, \quad \text{sampled version: } x(n) = e^{2\pi i f_1 nT}, \quad -\infty < n < \infty \quad (2)
\]

The spectrum of the sampled signal over the Nyquist interval is:

\[
X(\omega) = 2\pi \delta(\omega - \omega_1), \quad -\pi \leq \omega \leq \pi \quad (3)
\]

It consists of a single sharp spectral line at \(\omega = \omega_1\) and is replicated at multiples of \(2\pi\).

**Real Signal**
The spectrum of the sampled signal over the Nyquist interval is:

\[ X(\omega) = \pi \delta(\omega - \omega_1) + \pi \delta(\omega + \omega_1), \quad -\pi \leq \omega \leq \pi. \]

The windowed sinusoid consists of \( L \) samples:

\[ x_L(n) = e^{j\omega_0 n}, \quad n = 0, 1, \ldots, L - 1 \]

Its spectrum can be obtained by using Eq. (3) in Eq. (1):

\[ X_L(\omega) = \int_{-\pi}^{\pi} x(\omega') W(\omega - \omega') d\omega' = \frac{1}{2\pi} \int_{-\pi}^{\pi} 2\pi \delta(\omega' - \omega_1) W(\omega - \omega') d\omega' = W(\omega - \omega_1) \]

The windowing process causes a smearing effect on the sharp spectral line \( \delta(\omega - \omega_1) \) at \( \omega_1 \) and replacing it with \( W(\omega - \omega_1) \) as shown in Fig. 4.4(a).

**Case 2: Two Sinusoids**

A linear combination of two complex sinusoids with frequencies \( f_1 \) and \( f_2 \) and amplitudes \( A_1 \) and \( A_2 \) is used for analysis. The analog, sampled, and the windowed signals with its spectra are as follows:

\[ x(t) = A_1 e^{2\pi j f_1 t} + A_2 e^{2\pi j f_2 t}, \quad -\infty < t < \infty, \]

Sampled version: \( x(n) = A_1 e^{2\pi j f_1 nT} + A_2 e^{2\pi j f_2 nT}, \quad -\infty < n < \infty \)

Spectrum of the sampled signal: \( X(\omega) = 2\pi A_1 \delta(\omega - \omega_1) + 2\pi A_2 \delta(\omega - \omega_2), \quad -\pi \leq \omega \leq \pi \)

Spectrum of the windowed signal: \( X_L(\omega) = A_1 W(\omega - \omega_1) + A_2 W(\omega - \omega_2) \)

It can be observed that the two sharp spectral lines are replaced by their smeared version as shown in Fig. 4.4.
3) Resolvability Condition [43][48]

The frequency separation, $\Delta f = |f_2 - f_1|$ or $\Delta \omega = |\omega_2 - \omega_1|$ in the example shown in Fig. 4.4 is large enough so that the mainlobes are distinct and do not overlap. But, the mainlobes of the two signals will start merging when $\Delta f$ approaches the mainlobe width $\Delta f_w$. Thus, the resolvability condition can be defined as $\Delta f \geq \Delta f_w = f_s / L$. This means the two sinusoids will appear as two distinct ones only if their frequency separation is greater than the mainlobe width. Also, the minimum number of samples required to achieve a desired frequency resolution $\Delta f$ can be found out from the following relationship:

$$L \geq f_s / \Delta f$$

From the above discussion it can be concluded that achievable frequency resolution is determined by mainlobe width. And, the sidelobes are the undesirable artifacts of the windowing process. They could be confused with the mainlobes of a weaker signal if left unsuppressed. The sidelobes can suppressed by using a non-rectangular window. A comparison of rectangular and non-rectangular windows in time and frequency domains
is depicted in Fig. 4.5. Non-rectangular windows in general have a shorter and wider mainlobe compared to the rectangular window. This in turn reduces the achievable frequency resolution of the windowed spectrum. The effective width of the mainlobe for any type of window is inversely proportional to the window length:

$$\Delta f = c \left( \frac{f_s}{L} \right) \quad \text{or} \quad \Delta \omega = c \left( \frac{2\pi}{L} \right)$$

in radians per sample

where “c” is a constant ($\geq 1$) which depends on the window used. The rectangular window has the narrowest width, corresponding to $c = 1$.

4) Categories of Non-Rectangular Windows [43]

There are two main categories of non-rectangular windows: 1) Fixed window functions like Hanning, Hamming and Blackman, 2) Adjustable window types like Chebyshev and Kaiser.
a) Fixed Window Functions

The three non-rectangular fixed window functions considered for discussion are Hann, Hamming and Blackman.

Fig. 4.6. Time and frequency domain plot of Hann window.

\[ Hann: \quad w[n] = \frac{1}{2} \left[1 + \cos \left( \frac{2\pi n}{2M + 1} \right) \right], \quad -M \leq n \leq M. \]

Its spectra from Matlab simulation is shown in Fig. 4.6. The mainlobe width is 13.428 MHz and the relative sidelobe attenuation is -31.5 dB.

\[ Hamming: \quad w[n] = 0.54 + 0.46 \cos \left( \frac{2\pi n}{2M + 1} \right), \quad -M \leq n \leq M. \]

Its spectra from Matlab simulation is shown in Fig. 4.7. The mainlobe width is 12.207 MHz and the relative sidelobe attenuation is -42.7 dB.
Blackman: \[ w[n] = 0.42 + 0.5\cos\left(\frac{2\pi n}{2M + 1}\right) + 0.08\cos\left(\frac{4\pi n}{2M + 1}\right), \quad -M \leq n \leq M. \]

Its spectra from Matlab simulation is shown in Fig. 4.8. The mainlobe width is 15.869 MHz and the relative sidelobe attenuation is -58.1 dB. The different properties of fixed window functions have been compared in Fig. 4.9 and Table 4.1.

b) Adjustable Window Functions

Fig. 4.8. Time and frequency domain plot of Blackman window.
Fig. 4.9. Spectra comparison of Fixed non-rectangular windows

**Chebyshev:**

\[ w[n] = \frac{1}{2M + 1} \left[ \frac{1}{\gamma} + 2 \sum_{k=1}^{M} T_k \left( \beta \cos \frac{k \pi}{2M + 1} \right) \cos \frac{2nk \pi}{2M + 1} \right], -M \leq n \leq M, \]

where \( \gamma \) is the relative sidelobe amplitude, \( \beta = \cosh \left( \frac{1}{2M} \cosh^{-1} \frac{1}{\gamma} \right) \) and \( T_i(x) \) is the \( i \)-th order Chebyshev polynomial. Its spectra from Matlab simulation is shown in Fig. 4.10.

The mainlobe width is 12.207 MHz and the relative sidelobe attenuation is -42.7 dB.

**Kaiser:**

\[ w[n] = \frac{I_0(\beta)}{I_0(\beta)} \left\{ \beta \sqrt{1 - (n/M)^2} \right\}, -M \leq n \leq M, \]

Where \( \beta \) is an adjustable parameter and \( I_0(u) \) is the modified zeroth-order Bessel function. Parameter \( \beta \) is computed from

\[
\beta = \begin{cases} 
0.1102(\alpha_s - 8.7), & \text{for } \alpha_s > 50, \\
0.5442(\alpha_s - 21)^{0.4} + 0.07886(\alpha_s - 21), & \text{for } 21 \leq \alpha_s \leq 50, \\
0, & \text{for } \alpha_s < 21.
\end{cases}
\]
Its spectra from Matlab simulation is shown in Fig. 4.11. The mainlobe width is 10.986 MHz and the relative sidelobe attenuation is -30.1 dB.

The different properties of adjustable window functions have been compared in Fig. 4.12 and Table 4.1. The compensation is applied along different window functions (fixed and adjustable) and the maximum achievable IDR of the receiver is compared in the fifth column of Table 4.1.

<table>
<thead>
<tr>
<th>Window function (n = 256, Fs = 2500 MHz)</th>
<th>3 dB Mainlobe width (MHz)</th>
<th>Relative sidelobe attenuation (dB)</th>
<th>Leakage (%)</th>
<th>Maximum two-signal IDR achieved with compensation (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rectangular</td>
<td>8.545 MHz</td>
<td>-13.3 dB</td>
<td>9.15</td>
<td>18</td>
</tr>
<tr>
<td>Kaiser (β = 4.088)</td>
<td>10.986 MHz</td>
<td>-30.1 dB</td>
<td>0.11</td>
<td>24</td>
</tr>
<tr>
<td>Hamming</td>
<td>12.207 MHz</td>
<td>-42.7 dB</td>
<td>0.03</td>
<td>23</td>
</tr>
<tr>
<td>Chebshev (γ = 45)</td>
<td>12.207 MHz</td>
<td>-45.0 dB</td>
<td>0.29</td>
<td>23</td>
</tr>
<tr>
<td>Hann</td>
<td>13.428 MHz</td>
<td>-31.5 dB</td>
<td>0.05</td>
<td>22</td>
</tr>
<tr>
<td>Blackman</td>
<td>15.869 MHz</td>
<td>-58.1 dB</td>
<td>0.00</td>
<td>15</td>
</tr>
</tbody>
</table>
5) **Frequency Resolution and Windowing**

The digital receiver design uses a 256-point FFT with no overlap. This means that the continuous signal is divided into several 256-sample slices as shown in Fig. 4.13. The point at which each slice begins and ends is arbitrarily determined by FFT frame size, and therefore it is unlikely to have an integer number of cycles in a given data set. This non-periodicity could lead to glitches and spurious responses in the frequency analyses.

![Time and frequency domain plot of Chebyshev window.](image)

Data windowing reduces the end-effect glitches and introduces a well-behaved pseudo-periodicity. In the previous section (Table 4.1), some of the commonly used fixed and adjustable windows were compared for mainlobe width, sidelobe suppression and maximum achievable IDR. It can be observed that even though Blackman window has the highest sidelobe attenuation (-58.1 dB) among the window functions, it did not achieve the maximum IDR. This is because, its mainlobe width is close to 16 MHz, much
higher than the frequency resolution of the receiver (10 MHz). If the mainlobe width of
the strong signal is larger than the frequency resolution, the mainlobe would spread to the
next frequency bin where a weak signal may be present, and cause a false alarm. Thus, a
window function with a small mainlobe width, ideally close to the frequency resolution
(10 MHz), and good sidelobe suppression is recommended.

Out of the five window functions considered, the Kaiser window function has a
mainlobe width closest to the frequency resolution of the receiver and had a good
sidelobe suppression. And, therefore the receiver configuration using Kaiser window
function along with compensation achieved the maximum two signal IDR of 24 dB.

![Fig. 4.11. Time and frequency domain plot of Kaiser window.](image)

In the next section, the effectiveness of using Kaiser Window and compensation
method for extension of the two-signal IDR of wideband digital receivers is discussed. A
novel hardware implementation of the Kaiser Window and the compensation method
using an arithmetic scheme comprised of fixed-point and floating-point representations is presented.

![Fig. 4.12. Spectra comparison of adjustable non-rectangular windows](image)

4.3 Receiver-on-a-chip II (ROC II)

A 1-GHz signal bandwidth digital receiver, called receiver-on-a-chip (ROC) II, is divided into two areas: 1) signal sampler and formatting, and 2) windowing/super-resolution and frequency measurement, as shown in Fig. 4.14. The signal sampler comprises of a 2.5-GHz sampling rate ADC which samples the signal every 0.4 ns to
produce a 4-b amplitude measurement. Each bit from the ADC is then passed to 4 1:16 window samplers which convert the serial data stream to parallel and down convert the data rate by a factor of 16 to meet the speed at which the detection circuit can receive data, i.e., \((2.5\text{-GHz sampling rate})/(16\text{-sample window}) = 156.25\text{ MHz sampling rate}.

![Fig. 4.14. Two areas of ROC II](image)

As depicted in Fig. 4.14, the windowing/super-resolution and frequency measurement system has four main blocks: 1) Window Function, 2) FFT, 3) Super-Resolution, and 4) Frequency Selection Logic. As to the design of the window function block, some of commonly used fixed and adjustable non-rectangular windows were compared in the previous section, mainly for three parameters: mainlobe width, sidelobe suppression and maximum achievable IDR (Table 4.1). Out of the five window functions considered, the Kaiser window function has a mainlobe width close to the frequency resolution of the
receiver (10 MHz) and a good sidelobe attenuation. It can achieve a two signal IDR of 24 dB.

![Architecture for the Kaiser Window block.](image)

The proposed Kaiser Window function design contains 256 look–up tables (LUTs), as shown in Fig. 4.15. The LUT stores the multiplication products of the Kaiser coefficients and 16 possible 4-b ADC outputs (from 0000 to 1111). By reading in the 4-b ADC output, its multiplication product with the Kaiser Window coefficient is read out from the LUT. The 256 LUT outputs are the inputs to 256 points FFT. The value for \( \beta \) to generate the Kaiser Window function coefficients was empirically chosen (\( \beta=4.088 \)) to provide the best performance. Other than the use of the Kaiser Window function, the ROC II uses a 12-point kernel function to implement the 256 points FFT. In comparison with the 4-point kernel function of Monobit receiver [8] ROC II shows a remarkable improvement of two-signal IDR.

The super-resolution is a compensation method whereby a stored pre-calculated response that closely approximates the highest amplitude frequency \( (f_1) \) response is
subtracted from the actual response. Nearly complete sidelobe cancellation can be achieved with this method provided the approximation is good enough. Only a narrow null window needs be used in this case; in fact, if the approximation is accurate enough, no window is even necessary. In the compensation method we assume that all significant sidelobes and spurs are generated by \( f_i \) alone. A complete sidelobe cancellation can be achieved if we can regenerate \( f_i \) correctly and subtract it from the actual response. This in turn can expose the weak second signal which otherwise could have been undetected in the midst of signal sidelobes and noise spurs. In our method, the pre-calculated response that closely approximates the \( f_i \) response is pre-stored in a high-resolution compensation matrix. The higher resolution \( f_i \) is then used to determine the correct row of compensation matrix for subtraction from the actual response. In the frequency selection logic, signal is compared with pre-determined threshold and based on the amplitude of the signal it is declared as a detected signal or a noise spur.

4.4 Kaiser Window and compensation

1) Effect of normalization on compensation

Fig. 4.16 shows the effect of compensation with and without normalization of the FFT outputs. Fig. 4.16 (a-b) shows the FFT output data before compensation operation. The data width of the FFT outputs are 8 bits wide and therefore amplitude ranges are \((0 – 256)\) and \((0 – 1)\) for unnormalized and normalized data respectively. The Fig. 4.16 (c-d) shows the strong signal data retrieved from the compensation LUT. In Fig. 4.16 (e) shows the result of compensation without normalization of FFT outputs. For an efficient compensation operation, the range of data being compensated should match the data.
range of the strong signal stored in the compensation look-up table (LUT). Otherwise, unbalanced compensation takes place and high amplitude strong signal spurs left behind in the processed data after compensation as shown in Fig. 4.16 (e). These high amplitude spurs would pose as real signals and cause false alarms, degrading the attainable two signal instantaneous dynamic range (IDR) of the receiver. In Fig. 4.16 (f), the result of the compensation with normalization is shown. It is seen that when compensated data and strong signal data have the same data range, an efficient compensation can be performed which exposes the weak signals.

Fig. 4.16. Effect of normalization on compensation

2) **Effectiveness of Kaiser Window function and compensation**

A study with a simulation example that underlines the effectiveness of the Kaiser Window function and compensation to improve the two-signal IDR is presented. Fig.
4.17(a) shows the frequency response after FFT in which the peak signal frequency $f_1 = 797$ MHz and the weak signal frequency $f_2 = 712$ MHz and the weak signal $f_2$ is $24$ dB down of the peak signal $f_1$. The response shows the sidelobes and spurs generated from the peak signal $f_1$. In the application of the Kaiser Window function as shown in Fig. 4.17(b), the mainlobe width is reduced to $10$ MHz and the sidelobes have good attenuation. In the application of the compensation as shown in Fig. 4.17(c), sidelobes and spurs are effectively cancelled. But neither approach of Fig. 4.17(b) and 4.17(c) can expose the second signal $f_2$. For both cases a spurious signal (which is above the threshold),

![Signals before Compensation and Windowing](image1)

![Signals after Windowing only](image2)

![Signals after Compensation only](image3)

![Signals after Compensation and Windowing](image4)

Fig. 4.17. The effectiveness of Kaiser Window and compensation to expose the weak signal (712 MHz) from the frequency response: (a) Without Kaiser Window function and compensation, (b) Using the Kaiser Window, (c) Using compensation, (d) Using both the Kaiser Window function and compensation.
instead of the actual signal $f_2$, was detected as the second signal. The receiver reports a false alarm. However, combining both Kaiser Window function and compensation, the sidelobes and spurs which could have caused false alarms, are effectively removed. The actual second signal $f_2$ is then exposed and detected as shown in Fig. 4.17(d).

3) **Hardware Design and Implementation**

a) **Kaiser Window Function**

The hardware of the Kaiser Window function was realized using 256 combinational look-up tables (LUTs) as shown in Fig. 4.15. Each combinational LUT stores 16 products from multiplication of its Kaiser coefficient and 16 possible 4-b digitized inputs (0000 – 1111). Each combinational LUT stores 16 products from multiplication of its Kaiser coefficient and 16 possible 4-b digitized inputs (0000 – 1111). An example of construction of the ‘Kaiser coefficient LUT 102’ is illustrated in Fig. 4.18. For our application the 102nd Kaiser coefficient of the window function is 0.93. It is multiplied with 16 possible numbers, ranging from 0 to 15. Each product is then quantized to one of the 16 levels and converted to a 4-b binary number. For example, the Kaiser coefficient 0.93 multiplied with 15 will produce a product of 13.95. The closest quantization level is 14. Therefore, the product after the window function is stored in the LUT by ‘1110’. Similarly, each of 256 Kaiser LUTs stores 16 4-b binary numbers. Moreover, each digitized 4-b ADC output is a 4-b input to its LUT and also serves as a 4-b address of the LUT. For example, if the 4-b digitized ADC output to the ‘Kaiser coefficient LUT 102’ is ‘1111’ (=15), by reading the address of ‘1111’ the output of the Kaiser LUT is ‘1110’ (=14).
**b) Super-Resolution Block**

The super-resolution block basically consists of four sub-blocks. They are: 1) Normalization, 2) Find “n” block (find the compensation row), 3) Compensation LUT, and 4) Compensation, as shown in Fig. 4.19.

The Normalization block takes in: 1) 128 FFT outputs (128 amplitudes and each is truncated to 8 bits) and 2) the highest amplitude among 128 outputs, “Max”. It normalizes the 128 amplitudes by dividing them with the “Max”. The Find ”n” block calculates the compensation row, based on the peak signal amplitude “Max” and its frequency bin of FFT. The Compensation LUT has a size of 1024x128 bit and the data is retrieved by using the 10-b address, the output of the Find ”n” block. The compensation block subtracts the compensation row data (1024-b), which closely approximates the highest peak signal “fl”, from the normalized FFT output (1024-b) for removal of the sidelobes and spurs of “fl”. The second weak signal is then exposed.
(1) **Normalization**

The Normalization block comprises of four main sub-blocks, as shown in Fig. 4.20. They are: 1) Multiplier block, 2) $1/X_{\text{max}}$ LUT block, 3) Control logic block, and 4) Tristate Buffer block.

(a) **Multiplier Block**

In theory, the fastest implementation of a binary multiplier would be a look-up table (LUT) with the products of the multiplier and the multiplicand pre-stored. No actual arithmetic operation is involved. This approach ceases to be practical as the operand size approaches 5 bits. But, a practical fast multiplier can be implemented by using smaller LUTs and few adders by adopting the split-multiplier approach [49]. This approach uses...
larger operands to generate partial products, and thereby reduces the number of partial products that needs to be summed to find the final product.

A novel LUT multiplier LUT is presented for the fixed-point multiplication. To retrieve the partial products, the split multiplier and the multiplicand are concatenated to
form an address of the LUT where the partial products are stored. The partial products of the split operands are then summed after appropriate interleave shifting.

In Fig. 4.21, this principle is illustrated using a 4-digit decimal multiplier and multiplicand. The four digit operands are split into two digit multiplicands and two digit multipliers which result in four operands. The four partial multiplication operations are performed in parallel. The results of the first two multiplications and the last two multiplications are added in parallel with necessary shifting. The partial addition results are finally summed after appropriate relative bit shifting. In a similar manner, fast implementation of binary multipliers involving large operands can be realized by splitting the multiplier and multiplicands, and using LUTs to store partial products of larger operands. This reduces the number of additions required, and ultimately reduces the time required to find the binary product. A general flow of the LUT multiplier is explained in Fig. 4.22.
In the first step, product data of the partial multiplication is retrieved from the LUT. The number of individual LUT required in this stage is dependent on the size of the operands involved in the multiplication. The partial product retrievals from LUTs are done in parallel. In the next stage, the partial products are summed using two adders, which compute the summation in parallel. In the final stage, the outputs from two adders in the previous stage are summed together to obtain the final product. The required time to compute the final product is the sum of the time to retrieve the partial products from the LUT, and the time it takes for the data to traverse through the first and second stage adders.

Fig. 4.23. LUT construction for 4 x 4 Multiplication

(i) Construction of Product LUT
In the LUT multiplier architecture described in the previous section, we use only one basic LUT, a 256 x 8 bit LUT for a 4 x 4 Multiplication. The LUT construction is described in Fig. 4.23. It has 6 rows to accommodate all combinations of 4 x 4 multiplications and each row has 8 bits of the pre-calculated 4 x 4 multiplication result. The eight address bits for the LUT are arranged in such a way that the first four bits represent the multiplicand and the last four bits represent the multiplier. The corresponding product is stored in the row whose address is a concatenation of the multiplier and the multiplicand. For example, in Fig. 4.23, “01010100” is the product of “0111” and “1100”. It is stored in the address 115 (“01111100”) so that to retrieve the product result, we only need to concatenate the multiplicand and the multiplier together and use it as the address for the LUT.

(ii) Operation of the LUT Multiplier

In Fig. 4.24, a pipelined 8 x 8 multiplier is described. It mainly comprises of four 256 x 8 bit LUTs and three binary adders of which two are 8-b adders and one is a 12-b adder. The LUTs store pre-calculated multiplication products. Once the partial products are retrieved from the LUTs, they are summed together by appropriate interleave shifting. In this example, the multiplicand (MD) is “01111011” and the multiplier (MR) is “10000101”. At the first stage, the MR and the MD are split to MR₇₋₄ (1000 x 2⁴), MR₃₋₀ (0101), MD₇₋₄ (0111 x 2⁴) and MD₃₋₀ (1000). The corresponding split operands is then concatenated into 8-bit addresses to retrieve the pre-calculated 4x4 multiplication results from the LUTs. The concatenated addresses of “01111000”, “10111000”, “01110101” and “10110101” retrieve pre-calculated products of “00111000”, “01011000”, “01011000”,
“00100011” and “00110111”, respectively. The partial products are then summed by appropriate interleave shifting using two 8-b adders and finally the results of the adders are summed using a 12-b adder.

Fig. 4.24. Pipelined 8 x 8 multiplier.
Fig. 4.25 describes the architecture for a 16x16 multiplier with a detailed example. In the 16x16 multiplication, the partial products of the split operands are computed using 8x8 LUT multipliers. The performance comparison of the proposed multiplier is given in Table 4.2.

(iii) General N x N Multiplication

In this section an NxN multiplier with multiplicand (MD [N-1:0]) and multiplier (MR [N-1:0]) bits ranging from [N-1:0] is described.
The operands are split as follows

MD [N-1:N/2], MR [N-1:N/2], MD [N/2-1:0], MR [N/2-1:0].

The four partial products from the LUTs are

1) \((MD [N-1:N/2] \times 2^{N/2}) \times (MR [N-1:N/2] \times 2^{N/2})\)

\begin{align*}
= & (MD [N-1:N/2] \times MR [N-1:N/2]) \times 2^N \\
= & \gg MDMR1[N-1:0] \times 2^N
\end{align*}

2) \((MD [N/2:0]) \times (MR [N-1:N/2] \times 2^{N/2})\)

\begin{align*}
= & (MD [N/2:0] \times MR [N-1:N/2]) \times 2^{N/2} \\
= & \gg MDMR2[N-1:0] \times 2^{N/2}
\end{align*}

i. \((MD [N-1:N/2]) \times (MR [N/2:0])\)

\begin{align*}
= & (MD [N-1:N/2] \times MR [N/2:0]) \times 2^{N/2} \\
= & \gg MDMR3[N-1:0] \times 2^{N/2}
\end{align*}

ii. \((MD [N-1:0]) \times (MR [N/2:0])\)

\begin{align*}
= & (MD [N-1:0] \times MR [N/2:0]) \\
= & \gg MDMR4[N-1:0]
\end{align*}

**Partial product adder (PPA) stage**

*Adder 1:*

\((MDMR1[N-1:0]) \times 2^N + (MDMR2[N-1:N/2]) \times 2^N =\)

\((MDMR1[N-1:0] + MDMR2[N-1:N/2]) \times 2^N = \gg PPA\_OUT[N-1:0] \times 2^N\)

*First output from the partial product adder:*

\{\(PPA\_OUT[N-1:0] \times 2^N, MDMR2[N/2:0]) \times 2^{N/2}\}\)

\(= \gg (F\_ADDER\_IN1[3N/2 \times 1:0]) \times 2^{N/2}\)
Adder2:

\[(MDMR3[N-1:0]) \times 2^{N/2} + (MDMR4[N-1:N/2]) \times 2^{N/2} = \]

\[(MDMR3[N-1:0] + MDMR4[N-1:N/2]) \times 2^{N/2} = \]

\[=> (F\_adder\_in2 [N-1:0]) \times 2^{N/2} \]

Final adder stage

\[(F\_adder\_in1[3N/2 -1:0]) \times 2^{N/2} + \]

\[(F\_adder\_in2 [N-1:0]) \times 2^{N/2} = \]

\[F\_adder\_out[3N/2 -1:0]) \times 2^{N/2} \]

Final product output  = >>

\{F\_adder\_out[3N/2 -1:0] , MDMR4[N/2-1:0] \}

(b) 1/X_{max} LUT

(i) Bit Format of the Inverted Element Stored in LUT

Inverse of each element in the required bit range is pre-calculated and then converted to a binary number. To store precision digits in the LUT, the binary point of the resulted binary number is shifted right until a predetermined number of non-zero binary digits are obtained. The number of shifts (bias bits) is then stored in the LSB and the inverted number in binary is stored in the MSB. The inverted numbers are stored in the LUT in a way that the highest element can be used as the address to the LUT for the data retrieval. Fig. 4.26 explains how to determine and store an inverse of a binary number. For example the inverse of 123 (1011001B) is taken and it results in 0.008130081.. (0.00000010000101B). Not much useful information of the inverted number is obtained if only the first eight bits after the binary point is stored in LUT.
Table 4.2: Performance comparison of the proposed LUT Multiplier

<table>
<thead>
<tr>
<th>Multiplier Design</th>
<th>Area</th>
<th>Delay(ns)</th>
<th>Power Dissipation (mW)</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>[50]</td>
<td>16x16</td>
<td>-</td>
<td>52.6</td>
<td>-</td>
</tr>
<tr>
<td>[51]</td>
<td>16x16</td>
<td>12,349 trans</td>
<td>7.4</td>
<td>33.7</td>
</tr>
<tr>
<td>[52]</td>
<td>8x8</td>
<td>70.74 µm x 67.52 µm</td>
<td>6.98</td>
<td>0.336@100MHz z</td>
</tr>
<tr>
<td>[53]</td>
<td>8x8</td>
<td>4812 transistors</td>
<td>1</td>
<td>100.52</td>
</tr>
<tr>
<td>[54]</td>
<td>16x16</td>
<td>9.03 mm²</td>
<td>3.98</td>
<td>-</td>
</tr>
<tr>
<td>Proposed design</td>
<td>8x8</td>
<td>9,874 transistors</td>
<td>0.92</td>
<td>2.16590</td>
</tr>
<tr>
<td></td>
<td>16x16</td>
<td>42,646 transistors</td>
<td>1.65</td>
<td>3.67260</td>
</tr>
</tbody>
</table>

To get effective representation, the binary point is shifted fourteen times to the right so that we obtain 8 digits of useful information to store in the LUT. The bits that would be finally stored (“10000101“) is concatenated with the bias bits “1110”. And the resulted binary number “10001011110” is evaluated as follows: \((2^7 + 2^2 + 2^0) \times 2^{-14} = 0.008117675\).

(ii) LUT Construction for \(1/X_{\text{max}}\)

The data for the required range of numbers is prepared as explained in the previous section. The data is stored in the LUT in such as way that the highest number among the data set, \(X\), is used as the address for retrieving the inverse \((1/X_{\text{max}})\) from the LUT. This reduces the hardware in terms of special logic required to select the inverse from the
LUT. In Fig. 4.27, the range of numbers considered for the LUT is from 1 to 128. For example, 123 (7BH) is a number to be stored and 10001011110 is the binary of its inverse format. Thus the required size of the LUT is 128x12 bits, requiring seven address bits for data retrieval. The inverse of the number 123 (7BH) is stored in the 123rd row of the LUT so that seven bits (1011001) of the number itself can be used as the address for retrieving the data corresponding to the inverse from the LUT. Given a dividend is “00011001” and a divisor is “00001001”. The divisor is used as the address to the LUT to retrieve the inverted value of the divisor. The output of the LUT is a 12-bit binary number “111000111011”. This is then split as data (“11100011”) and bias bits (“1011”). The data (inverted divisor) and the dividend is multiplied using the 8x8 LUT multiplier.
described in the earlier section. The 16-bit product is then combined with the bias bits using a combinational block to produce the result of the division in an 0.8 fixed point format. An example of normalization is given in Fig. 4.28.

\[
X_n \times \left( \frac{1}{X_{\text{max}}} \right) = 0.5691056 \ldots
\]

\[
\begin{array}{c}
\text{Bias bits} \\
\text{Inverted } X_{\text{max}} \\
\text{1/ } X \text{ LUT} \\
\end{array}
\]

\[
\begin{array}{c}
X_{\text{max}} \\
\text{8} [01111011] \\
\text{8} [01000110] \\
\end{array}
\]

\[
\begin{array}{c}
X_n \\
\text{8} [01000110] \\
\end{array}
\]

\[
\begin{array}{c}
[01000110] \times [01000110] \\
8 \times 8 \text{ binary multiplication} \\
\end{array}
\]

\[
\begin{array}{c}
[0010010001011110] \\
\text{16} \\
\text{Convert the product into 0.8 fixed-point format} \\
\end{array}
\]

\[
[0.10010001] = 0.56640625
\]

Fig. 4.28. Example of normalization

(c) Control Logic
This is a combinational block, which generates signal pulses based on the input clock. The generated pulses in turn serve as the enable signals of the Tristate Buffer block and the output latches of the Normalization block.

(d) Tristate Buffer Block

The Tristate Buffer block functions as multiplexers. There are a total of 1024 bits to be multiplied in the Multiplier block. The Multiplier block can take in only 64 bits at a time. So, initially the first 64 bits are taken in for multiplication and after the operation is completed the next 64 bits are taken in. This sequential multiplication continues until all 1024 bits are multiplied. This block contains 1024 non-inverting tristate buffers, connected to the 1024 input bits. The 1024 tristate buffers are divided into groups of 64, and thus 16 groups in total to take in 1024 bits. A common enable signal is applied to 64 buffers in the same group. Thus the block requires 16 enable signals in total. The 64 output bits of the Tristate Buffer block are inputs to eight LUT multipliers shown in Fig. 4.20.

b) Find “n” Block (find the compensation row)

The sensitivity of the calculated row number that closely approximates the highest amplitude signal \(f_1\) in the compensation LUT is presented. In the proposed method the pre-calculated response that closely approximates the \(f_1\) response is pre-stored in a high-
Fig. 4.29. Block diagram for Find “n” block

resolution compensation matrix. After estimating the amplitude and frequency of the highest amplitude signal $f_1$, the row “n” in the compensation matrix that closely approximates the highest amplitude signal $f_1$ is then calculated using the super-resolution algorithm. In the super-resolution algorithm this row number “n” is estimated based on the two neighboring bins of $f_1$. 
Fig. 4.30. The effect of compensation with different row selection.

Table 4.3: The Sensitivity of Row Estimation in Compensation Matrix for 10,000 Simulation Runs

<table>
<thead>
<tr>
<th>Random error added to “n”</th>
<th>Second signal detection (%)</th>
<th>Second signal miss (%)</th>
<th>Second signal false alarm (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>{0}</td>
<td>81.94</td>
<td>17.35</td>
<td>0.71</td>
</tr>
<tr>
<td>{-1,0,1}</td>
<td>76.58</td>
<td>14.47</td>
<td>8.95</td>
</tr>
<tr>
<td>{-2,-0,1,2}</td>
<td>60.35</td>
<td>11.01</td>
<td>28.64</td>
</tr>
</tbody>
</table>
The inputs to this block are the highest amplitude “a” among the 128 outputs of the FFT block, amplitudes of its right and left immediate neighbors (“b” and “c”), the frequency bin of the highest amplitude (“m”) and its neighboring frequency bins (“m+1” and “m-1”) as shown in Fig. 4.29. Using the super-resolution algorithm, the row “n” in the compensation matrix that closely approximates the highest amplitude signal \( f_i \) is then calculated. The output of the block is a 10 bit binary number “n”, the compensation row.

Before the row estimation of “n” is discussed, Table 4.3 reports the sensitivity of “n” by deliberately adding an error to the calculated “n”. An error of \{-1,0,1\} randomly added to “n” reduces the second signal detection to 76.58% from 81.94% and increases the second signal false alarm to 8.95% from 0.71%. Whereas an error of \{-2,-1,0,1,2\} even further reduces the second signal detection to 60.35% and increases the second signal false alarm to 28.64%.

Fig. 4.30 demonstrates this effect of compensation method with different compensation rows, ranging from n-4 to n+4 where n is the value of calculated compensation row from the super-resolution algorithm. It is observed that the compensation using the calculated compensation row value “n” by the proposed super-resolution algorithm nearly completely eliminates the sidelobes and spurs from the original signal and thus exposes the second signal. Whereas the compensation using the compensation row other than “n” does not eliminate sidelobes and spurs effectively.

c) Compensation LUT
The Compensation LUT block has 1024 x 128 elements, each is 8-b wide as shown in Fig. 4.31. The 8-b width was chosen after comparable simulation results before and after bit truncation. The estimated compensation row number “n” is used as the 10-b address of the Compensation LUT. This Compensation block is basically 128 parallel subtractors which subtract the pre-stored values of approximated highest amplitude frequency (f1) response from the actual response. The inputs to the block are two 1024-b operands from the normalized FFT and the Compensation LUT, and the output is 1024 bit wide, as shown in Fig. 4.19.

4.5 Timing of Super-Resolution Block

The Find-Compensation-row and Normalization blocks are operated in parallel. The timing diagram of the Super-Resolution is shown in Fig. 4.32. The normalization and find “n” blocks are completed in 13 and 23 clock cycles. Thus, the 10 bit address to the
compensation LUT to approximate the highest signal, and the normalized data values are ready at 20.8 ns and 36.8 ns respectively. The 10-b address is then used to retrieve the 1024-b highest signal data from the compensation LUT, which takes additional 2 clock cycles (3.2 ns). The compensation operation, which subtracts the highest signal from the input data, is completed in 38.4 ns.

![Fig. 4.32. Timing diagram of Super-Resolution block](image)

4.6 Simulation Results

Many simulations were conducted to demonstrate the effectiveness of the Kaiser Window function and the compensation method. In these simulations a white Gaussian noise of power -80.03 dBm is added to the input signals. Combinations of 2-b and 4-b ADC, 64 and 256 points FFT implemented using 4-point and 12-point kernel function, ideal 256 points FFT, Kaiser Window, and compensation were selected.
Table 4.4: Two-signal IDR Results

<table>
<thead>
<tr>
<th>2-signal SFDR(dB)</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>7</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>9</th>
<th>11</th>
<th>11.5</th>
<th>12</th>
<th>14</th>
<th>17</th>
<th>18</th>
<th>22</th>
<th>24</th>
<th>29</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-b ADC</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>4-b ADC</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>4-point kernel, 256-pt FFT</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12-point kernel, 64-pt FFT</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12-point kernel, 256-pt FFT</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ideal 256-pt FFT</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Kaiser Window</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Compensation</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
The strong signal frequency, $f_1$, was varied from 125 MHz to 1,125 MHz. The amplitude of the $f_2$ signal varies from 0 db to 30 db and the frequency varies from 125 MHz to 1,125 MHz. At a constraint of the weak signal $f_2$ missing less than 20% and the

Table 4.5: Transistor Count for each Subsystem

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Transistor count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock divider</td>
<td>4,275</td>
</tr>
<tr>
<td>Window sampler and data collector</td>
<td>46,850</td>
</tr>
<tr>
<td>Kaiser Window function</td>
<td>18,884</td>
</tr>
<tr>
<td>56 pt, 12 pt-kernel FFT</td>
<td>1,392,148</td>
</tr>
<tr>
<td>Super-Resolution block</td>
<td>550,276</td>
</tr>
<tr>
<td>Frequency selection logic</td>
<td>193,105</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>2,205,538</strong></td>
</tr>
</tbody>
</table>

Table 4.6: ROC II Statistics

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>130 nm CMOS</td>
</tr>
<tr>
<td>Transistor count</td>
<td>2,205,538</td>
</tr>
<tr>
<td>Die size</td>
<td>1.98 mm x 1.98 mm</td>
</tr>
<tr>
<td>Total I/O pins</td>
<td>62</td>
</tr>
<tr>
<td>Power supply</td>
<td>1.2 V</td>
</tr>
<tr>
<td>Clock rate</td>
<td>2.5 GHz</td>
</tr>
<tr>
<td>Input data rate</td>
<td>10 Gb/s</td>
</tr>
<tr>
<td>Output data rate</td>
<td>156 Mb/s</td>
</tr>
<tr>
<td>Power dissipation</td>
<td>1 W</td>
</tr>
</tbody>
</table>
false alarm less than 1%, 10,000 simulation runs were taken and each run generated two signals for simulation. Exhaustive simulation of different configurations of ADC, FFT, window function, and compensation for maximum achievable two-signal dynamic range of the receiver is tabulated in Table 4.4. It is shown that by using 4-b ADC and 256 points FFT of 12-point kernel function a maximum two-signal IDR of 9 dB is obtained. The IDR is extended to 14 dB by using the Kaiser Window, to 18 dB by using the compensation, and to 24 db by using the Kaiser Window and the compensation together.

Furthermore, by using 4-b ADC and ideal 256 points FFT a maximum two-signal IDR of 11 dB is obtained. The IDR is extended to 17 dB by using the Kaiser Window, to 22 dB by using the compensation, and to 29 db by using the Kaiser Window and the compensation together.

4.7 Noise and Bit Truncation Effect on Receiver Performance

To conduct a further study of the proposed digital receiver performance, possible noise and truncation errors are introduced to the receiver system at as shown in Fig. 4.33. As the front-end radio frequency (RF) interface to the wideband receiver, an amplifier with a pre-determined gain and noise figure is used before the ADC to match the input dynamic range to ADC. In this section, we study the effect of noise and bit-truncations on the two-signal IDR of wideband digital receivers. Three sources of noises are considered in this study: amplifier output noise \( (N_o) \), ADC integral non-linearity error (INL), and random noise error at the output of the ADC. The output bits are truncated at different stages to reflect the bit truncations in the actual hardware implementation.
Signal-to-Noise Ratio (SNR) is signal power in the receiver divided by mean noise power of the receiver. Incoming signal is not detectable if the signal power is less than or equals to the noise power. Thus, for a signal to be detected, the signal energy plus the noise energy must exceed a pre-determined signal detection threshold of the receiver. The threshold value is normally chosen high enough above the mean noise level so that the spurious signals possibly exceeding the threshold and causing false alarms will unlikely happen.

The amplifier output noise can be calculated as follows:

**Noise power, $N_o = N_i + B(\text{bandwidth}) + \text{NF(Noise figure)}$ dBm \( \quad \text{(1)} \) \[13\]

where $N_i = \text{noise power at the input of the amplifier per unit bandwidth (}-174 \text{ dBm})$, and is calculated as $kT$, where $k$ is the Boltzman constant and $T(290K)$ is the room
temperature. The bandwidth of the receiver under consideration is 1.25 GHz. The Noise figure of the amplifier is assumed as 3 dB [55]. Using these values in the above equation, the noise power is found as:

\[ (-174 + 10 \log(1.25e9) + 3) \text{ dBm} \]

\[ = (-174 + 90.9691 + 3) \text{ dBm} \]

\[ = -80.0309 \text{ dBm} \]

The noise power of white Gaussian noise \( N_0 \) in dBm calculated using Eq. (1) is converted to amplitude in Volt using following steps:

1) Since dBm represents the power level compared to 1 mW, the noise power in dBm has to be converted to Watt before it can be used to calculate the amplitude in Volt. The following formula is used to calculate noise power in Watts:

\[ \text{dBm} = 10 \log (P) \times (1000 \text{ mW/watt}) \quad --- (2) \]

where dBm = Power in dB referenced to 1 mW, and \( P \) = Noise power in W.

2) The noise amplitude in Volt is calculated using the formula:

\[ P = \frac{(V)^2}{R} \quad --- (3) \]

The Ohmic contact resistance \( R \) is assumed to be 50 \( \Omega \) [55]. Using Eq. (2) and (3), the noise power of the receiver is 2.2281e-005 Volt. A white Gaussian noise power \( N_0 \) is added to the input signal before the ADC stage.

2) ADC Integral Non-Linearity (INL) and Random Noise Errors
For $N = 4$ bits, a 1-V full-scale range is divided into $2^N = 16$ levels. The size of the least-significant bit (LSB) is $1/16$ V.

Fig. 4.34. Second signal false alarm considering ADC integral non-linearity (INL) and random noise errors up to 50% of LSB

In this section, ADC integral non-linearity (INL) and random noise errors are considered for false alarm analysis. Exhaustive simulations are performed to study the tolerance of the system to noise spurs that ultimately limits the performance of the receiver. It is observed from simulation shown in Fig. 4.34 and 4.35 that the second signal false alarm is less than 1% and 1.2% while considering ADC integral non-linearity (INL) and random noise errors up to 50% and 150% LSB, respectively. The second signal false alarm increase by 0.2% when ADC INL and random noise errors increase from 50% to 150% LSB. ADC INL and random noise errors have very limited impact on the false alarm.
Simulations were conducted for the receiver after adding amplifier output noise to the input signal, ADC integral non-linearity (INL) and random noise errors to the ADC and the output of the ADC respectively. The product of the ADC outputs and its corresponding Kaiser coefficients are quantized to 16 levels before it is passed to the FFT block. The outputs of the FFT block and strong signal data from the compensation look-up table (LUT) are truncated to 8-bits as in the hardware implementation.

10,000 signal cases were generated and simulated, and at different stages, amplifier output noise, ADC integral non-linearity (INL) and random noise errors are added to evaluate the performance of the receiver with the following constraints:

1. Minimum of 80% weak signal detection,
2. Maximum of 20% weak signal missing,
3. Maximum of 1% weak signal false alarm.
Table 4.7 gives the maximum achievable two signal IDR of the receiver for different levels of ADC integral non-linearity (INL) and random noise errors added to the ADC.

Table 4.7: Maximum achievable two signal IDR with consideration of amplifier output noise and bit truncation

<table>
<thead>
<tr>
<th>ADC integral non-linearity (INL) error vs. size of LSB (%)</th>
<th>Random noise error vs. size of LSB (%)</th>
<th>Maximum achievable two signal IDR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>23.5</td>
</tr>
<tr>
<td>50</td>
<td>50</td>
<td>20</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>16</td>
</tr>
<tr>
<td>150</td>
<td>150</td>
<td>14</td>
</tr>
</tbody>
</table>

Next, the verification of the implemented hardware within the software (Matlab) environment is discussed. This process involves several data conversions between different number representations (decimal and binary). Fig. 4.36 is a general block diagram demonstrating the overlap of the Matlab and HDL environment with data transfer between each other at different stages of the flow.

In the Matlab environment random signals are first generated and quantized and then is passed to the Kaiser Window function and the FFT processor. Since the HDL
environment requires binary numbers as input, the 128 FFT outputs and the highest among the FFT output is converted to 1024 (128 x 8 bits) and 8 binary bits in Matlab.

Fig. 4.36. Block diagram for hardware-software verification

Fig. 4.37. Detailed flow chart for hardware-software verification
environment. These then serve as inputs to HDL environment. In the HDL environment, the FFT outputs are normalized, compensation row in the compensation matrix calculated and retrieved and finally compensation is applied as discussed in the earlier sections. The output of the HDL environment is 1024 binary bits which represents 128 elements (each 8 bits) after applying compensation. 1024 binary bits is then converted back to decimal for the comparison and verification purposes in the HDL environment.

For comparison and verification of the results from the HDL environment, the decimal output of the FFT in the Matlab environment is normalized, compensated and second signal probabilities are calculated in parallel as shown in Fig. 4.37. The detection probabilities of both the Matlab and HDL environment are then compared and compared.

In the next section, hardware implementation flow and chip details for the receiver is presented.

4.8 Chip Implementation

The design was implemented using the IBM 130 nm 8SFG CMOS standard cell library. The design was coded in Verilog and its function verified using DAI Signalscan. Cadence BuildGates was used to perform logical synthesis and optimization. Cadence PKS was used to perform placement and routing.

The chip design is broken down into six major subsystems (Fig. 4.14). The chip contains 2,205,538 transistors and has a die size of approximately 1.98 mm x 1.98 mm. The entire operations in the Kaiser Window function, 256 point 12-point kernel FFT,
Super-Resolution block, and Frequency Selection logic is completed within 102.4 ns. The transistor count of each subsystem is given in Table 4.5. Collected radar data was tested for performance measurement and verification of the receiver. The design and performance statistics are summarized in Table 4.6.

4.9 Summary

The dynamic range of the receiver is ultimately limited by the sidelobes and spurs of the strong signal. A new technique of using the Kaiser Window function and the compensation method to extend the two-signal IDR of wideband digital receivers was proposed. A detailed comparative study of different fixed and adjustable windowing functions is conducted. An exhaustive study of configurations of ADC, FFT, window function, and compensation for maximum achievable two-signal dynamic range of the receiver is presented. For an input signal quantized to 4 bits the proposed technique has a potential of extending the two-signal IDR to 24 db using the kernel function FFT and 29 dB using the ideal FFT. Hardware implementation of the Kaiser Window and the compensation method using novel arithmetic structures were presented. Noise and bit truncation effect on the performance of the receiver, verification of the implemented hardware in the software, and chip implementation were also presented.
V. MULTIPLE SIGNAL DETECTION AND MEASUREMENT USING A CONFIGURABLE WIDEBAND DIGITAL RECEIVER

5.1 Introduction

Multiple signal detection is a highly desirable characteristic of wideband receivers. The bottlenecks of multiple signal detection are noisy signal environment, signal interferences, and computational complexity involved in a real-time implementation. The lack of a priori knowledge about the waveform of interest makes the design of modern wideband receivers with multiple signal detection capability even more challenging. In the past several multiple signal detection schemes have been proposed. Multiple signal detection using optimum filters in presence of noise was attempted in [33]. The scheme compensates for the match filters incapability to process more than two signals. It addresses the problem of deterioration in the receiver performance with the increase in the number of signals to detect. The detection problem for multiple signals embedded in noisy environment is treated as a multiple hypothesis test based on log-likelihood ratios [34]. A scheme based on the sequentially rejective Bonferroni test together with the nonlinear weighted least squares approach for estimation of the signal parameters under hypotheses and alternative is presented in [35]. Multiple signal detection using the atomic decomposition and the expectation maximization (EM) algorithms was proposed in [36].
The proposed approach addresses the need of advanced signal processing algorithms for counterbalancing increasing complexity of the communication and radar signals.

5.2 Digital Receivers

The instantaneous dynamic range (IDR) of digital receivers, a power ratio of the strongest and the weakest signals that can be properly detected simultaneously, is ultimately limited by the signal sidelobes and noise spurs that may easily produce false alarms especially in a multi-signal environment. Fig. 5.1 shows a false alarm B occurs in a noisy signal environment because it becomes the first detected signal for its amplitude is above the signal detection threshold and also higher than the amplitude of the actual signal A. A Monobit digital wideband receiver with a capability of detecting up to two signals was first presented in [8]. The processing speed of the receiver was improved by reducing the complexity of fast Fourier transformation (FFT). The Monobit receiver can simultaneously detect and process two signals in 1 GHz bandwidth and achieve a two-signal IDR of 5 dB. Analysis and evaluation of the receiver performance and limitation of the Monobit receiver was presented in [27][29]. The receiver’s limited capability for detection of more than two signals is addressed in [29].

In [2], an improvement to the Monobit receiver was presented for extension of the two-signal IDR to 18 dB with the second signal false alarm is less than 1%. The receiver design mainly consists of a 4-b ADC, a 256-point FFT implemented using a 12-point kernel function, and a super-resolution block. The super-resolution block implementing a compensation technique improves the instantaneous dynamic range of the receiver by subtracting the estimated peak (strong) signal frequency response and its spurs from the
actual received frequency response and thereby the second weak signal is exposed. By using this technique the pre-calculated FFT outputs of peak amplitude frequencies were stored in a look-up table (LUT). After spectral analysis of the incoming signals, the super-resolution algorithm is applied to estimate the peak signal frequency. The frequency response of the peak signal is then retrieved from the compensation LUT for subtraction from the actual response.

Fig. 5.1 False alarm due to noise.

Fig. 5.2. Block diagram of digital receiver (ROC II).
A technique of using a Kaiser window to reduce the spectral leakage by eliminating the discontinuities at the time window edges and using a compensation method to uncover the weak signal for extension of the two-signal IDR of wideband digital receiver to 24 dB was presented in [1]. The 1-GHz signal bandwidth digital receiver, called receiver-on-a-chip (ROC) II, was divided into two areas: 1) signal sampler and formatting, and 2) windowing/super-resolution and frequency measurement. The block diagram of ROC II is shown in Fig. 5.2. The signal sampler comprises of a 2.5-GHz sampling rate ADC which samples the signal every 0.4 ns to produce a 4-b amplitude measurement. Each bit from the ADC is then passed to 4 1:16 window samplers which convert the serial data stream to parallel and down convert the data rate by a factor of 16 to meet the speed at which the detection circuit can receive data.

In this section, a design reuse hardware configurable scheme of ROC II to achieve the multiple signal detection capability is presented. Performance evaluation of the configurable receiver to detect up to five signals and their maximum attainable IDR are presented.
5.3 Configurable Receiver

The proposed configurable receiver employs hardware reuse by effective configuration and detects multiple signals before the next set of buffered data arrives for processing. The block diagram of configurable receiver is shown in Fig. 5.3. The hardware blocks reused in the receiver are mainly the normalization, the frequency calculation, and the compensation blocks. In a real-time environment the number of signals is assumed to be unknown, i.e., the data received by the receiver may contain no signal (only noise), one signal or multiple signals. The received signal is first digitized by a 4-bit ADC. Fast Fourier transformation is then performed on the digitized data after the Kaiser Window function is applied. The data is then passed on to the subsequent blocks through the signal selector controlled by the control unit. The frequency of the detected strongest signal is calculated and stored after the data is normalized. This forms the first signal detection flow as depicted in Fig. 5.3. After that the next step is normalization which is a necessary step to compensate the strongest signal’s sidelobes and spurs. In the compensation operation, the signal with the highest amplitude and its sidelobes and spurs are subtracted and removed from the collected normalized data. The compensated data is then again passed on to the signal selector block. This procedure is repeated to form the $2^{nd}$, $3^{rd}$, ..., $n^{th}$ signal detection flow as depicted in Fig. 5.3. The control unit determines if a new set of radar data is passed on for new signal detection or the compensated data is passed on for continuation of detection of any existing subsequent signals in the previous set of radar data. Note every signal declared detected has to have its peak amplitude above the receiver detection threshold. This process will continue until a desirable maximum number of signals are detected.
A flowchart describing the control flow of multiple signal detection is depicted in Fig. 5.4. After the first signal is detected and processed, the control unit checks if the number of signals already processed reaches the maximum number of signal detection capability of the receiver. If the limit has not been reached, the processed data is selected to continue for processing. Otherwise, a new set of buffered radar data is selected to start a new signal detection process. The multiple signal detection procedure discussed above is illustrated by a case of five signal detection scheme shown in Fig. 5.5.
As shown in Fig. 5.5(a), the five signals are 223 MHz (the 1\textsuperscript{st} signal), 1092 MHz (the 2\textsuperscript{nd} signal), 1017 MHz (the 3\textsuperscript{rd} signal), 784 MHz (the 4\textsuperscript{th} signal) and 525 MHz (the 5\textsuperscript{th} signal). As part of the first signal detection flow, the 1\textsuperscript{st} signal (223 MHz) is detected and its frequency is then stored, followed by the first compensation operation. In the compensation operation, the strongest signal (herein the 1\textsuperscript{st} signal) is correctly estimated and removed from the data set along with its sidelobes and spurs which is shown in Fig. 5.5(b). The compensated data is then passed on to the signal selector controlled by the control unit. Next, either the compensated data (shown in Fig. 5.5(b)) or a new set digitized data is passed on for next level of signal detection. If the compensated data is passed on, the procedure for the 1\textsuperscript{st} signal detection and removal is repeated for detection
of each subsequent signal in an iterative fashion until the desirable maximum number of signals are detected. Figures 5.5(c), (d), and (e) depict the 3rd, the 4th and the 5th signal detected schemes.

5.4 Experimental Results

In a dense and noisy signal environment, it is often the strong signal sidelobes and spurs that interfere or overshadow the weak signals would cause false alarms. To achieve a realistic and accurate performance measurement, a worst case scenario is considered. For multiple signal detection scheme, except the strong signal all others are considered weak signals and their amplitude differences (dynamic range) from the strong signal are kept as high as possible. Fig. 5.6 illustrates the instantaneous dynamic range of five signal scheme. Even though the amplitudes of these weak signals are kept constant below of the strong signal in time domain, the signal amplitudes of the weak signals in
the frequency domain may be significantly different. This effect can be added to attributes of other undesired effects like ADC non-linearity errors and the spectral leakage in data windowing. And, it is often the weakest signal among the weak signals that ultimately limits the receiver performance especially in a multiple signal environment. In the following two, three, four and five signal detection cases, the signal with the second, the third, the fourth and the fifth lowest amplitude in the frequency domain is considered as the weakest signal among other signals. In this section only the simulation result of the weakest signal among all weak signals is discussed for its detection, miss and false alarm probabilities.

Simulations were performed for two, three, four and five signal cases. For each case, 10,000 signal cases were generated to evaluate the performance using the following receiver constraints:

1) A minimum of 80 % of the weakest signals are detected.
2) A maximum of 20 % of the weakest signals are missed.
3) A maximum of 1% of the detected signals are false alarm.

Fig. 5.7. Maximum attainable IDR for two, three, four and five signal cases
The maximum attainable IDR for two, three, four and five signal cases are 24, 21.5, 19.5 and 17 dB respectively which is depicted in Fig. 5.7. It is observed that the maximum attainable IDR of the receiver decreases with increase in the number of detected signals. Fig. 5.8 shows signal detection, miss, and false alarm probabilities of the receiver for the five signal case. It is observed:

1) Even though the signal amplitude of the weakest signals is 17 dB below the strongest signal, the signal detection, miss and false alarm for all signal cases are different. This can be attributed to undesired effects like ADC non-linearity and the spectral leakage in data windowing. The signal detection probability is decreased with increase in the number of signals detected.
2) The signal missing and false alarm probability is increased with increase in the number of signals to detect.

Fig. 5.9. Frequency error distribution for detected signals.

At a constraint of the weak signal miss less than 20% and the false alarm less than 1%, 10,000 simulation runs were taken and each run generated five signals for simulation. The distribution of frequency detection error for the five detected signals for 10,000 simulation runs is presented in Fig. 5.9. The frequency detection error between the detected signals and the actual signals is sorted based on the deviation from the actual frequency. The deviation ranges from a minimum of 1 MHz to a maximum of 10 MHz.
As seen from Fig. 5.9, about 85% error distribution is most likely concentrated in the low error bins like 1, 2, 3, and 4 MHz for all five cases.

5.5 Summary

Multiple signal detection is a highly desirable characteristic of a modern digital receiver. Its implementation is often a challenge due to a noisy signal environment, hardware and timing constraints in the realization of the design. A configurable digital receiver with multiple signal detection capability has been presented. The configurable scheme was discussed using the digital receiver, ROC II, by employing hardware reuse. Performance evaluation of the configurable ROC II receiver to detect up to five signals and their maximum attainable IDR has been presented.
VI. CONFIGURABLE AND EXPANDABLE FFT PROCESSOR FOR WIDEBAND COMMUNICATIONS

6.1 Introduction

A practical fast Fourier transform (FFT) processor can contain several millions of gates, so effective design techniques usually are required in order to guarantee high-speed products. A look-up table (LUT) methodology is developed and demonstrated on variable length (128–1024 point), variable bit-precision (6-12 b) FFT with uniform bit truncation and optimum bit truncation for wideband digital receiver in radar applications. The FFT processors are designed using a standard 130 nanometer CMOS process and operates down to 120 mV. The required processing time for the non-configurable 12-b 1024-point LUT FFT is 15.78 ns at a clock frequency of 470 MHz. The required time for configurable LUT 12-b 1024-point FFT processing is 61 ns. The configurable LUT FFT processor with short transform lengths can be expandable so that they can be used easily to form new FFT processors with longer transform lengths. The performance comparison of conventional FFT, LUT FFT, and configurable LUT FFT for digital wideband receiver application will be discussed.

The discrete Fourier transform (DFT) is widely used in design and implementation of digital signal processing systems. The DFT of a signal can be directly computed by Eq.
where \( N \) is the total number of sampled input points. The result is obtained from the product of \( x(n) \) and the kernel function \( e^{j\Phi} \).

\[
X (k) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi kn}{N}}
\]

A very efficient way to calculate DFT of a signal, based on the symmetry of the \( e^{-j2\pi nk} \) kernel was first proposed in [56], called fast Fourier transform (FFT). Since then several algorithms have been proposed to reduce the computational complexity of FFT implementation, including higher radix FFT [57], split-radix FFT [58], mixed-radix FFT [59] and an efficient FFT [60]. A dynamic scaling approach with data scheduling and pre-fetch buffering was presented in [61]. The throughput of the FFT processor was increased in [62] by using multi-path schemes. In [63] the number of complex multipliers is minimized via a bit-inverse and bit-reverse data scheduling scheme.

Kernel function FFT processors was proposed and adapted in a 1 GHz bandwidth monobit receiver design [8] to meet real-time (100 ns) processing requirements and to reduce hardware complexity by rounding the kernel function to predetermined 4 kernel points (+1, -1, +j, -j) by eliminating the multipliers and using only shifters and adders.

Because of the nonlinear nature of this approximation by the rounding errors, spurious responses are generated and limit the two-signal instantaneous dynamic range (IDR) to a level of 5 dB which is down by an average of 3~5 dB when an ideal FFT is used.

An improvement to the Monobit receiver was presented in Receiver-on-a-Chip (ROC) [2] for extension of the two-signal IDR to 18 dB. This receiver design is mainly
comprised of a 4-b ADC, a 256-point FFT implemented using a 12-point kernel function and a super-resolution block. In the 12-point kernel function, the DFT kernel function is replaced with an approximating kernel function of 12 values \((2, 2+j, 1+2j, 2j, -1+2j, -2+j, -2, -2-j, -1-2j, -2j, 1-2j, 2-j)\). The super-resolution block implementing a compensation technique improves the instantaneous dynamic range of the receiver by subtracting the estimated peak (strong) signal frequency response and its spurs from the actual received frequency response. Thereby the second weak signal is exposed. A technique of using a Kaiser window to reduce the spectral leakage by eliminating the discontinuities at the time window edges combined with the compensation method to further extends the two-signal IDR to 24 dB was presented in Receiver-on-a-Chip II (ROC II) [1]. Functional blocks of the digital receiver is shown in Figure 6.1.

A practical fast Fourier transform (FFT) processor can contain several millions of gates, so effective design techniques usually are required in order to guarantee high-speed products. A novel FFT processor which uses look-up tables to perform arithmetic operations is presented. Two types of FFT implementations, Kernel function FFT and conventional FFT, with uniform bit truncation and optimum bit truncation are studied and compared for their performance in wideband communications. An error analysis of frequency detection using the LUT FFT is presented. Gate count and delay of LUT FFT with and without configuration for 1024, 512, 256, and 128-point FFT are compared. The configurable LUT FFT processor with short transform lengths can be expandable to form new FFT processors with longer transform lengths.
6.2 Bit Truncation of Conventional FFT and Kernel function FFT

In a conventional FFT implementation, the kernel function $e^{j\Phi}$ values are stored in a ROM (Fig. 6.2). The values from the ROM is retrieved and passed on to a binary multiplier. A long transform length is usually the bottleneck of FFT speed. The next two sub-sections present digital receiver performance of using the conventional FFT and 12-point kernel FFT in which a long transform length is truncated to a short transform length.

Fig. 6.1. Functional blocks of the digital receiver [1].

1) Bit Truncation of Conventional FFT

Two schemes of bit truncation were applied to a 256-point radix-2 FFT. In the first scheme, all the stages were uniformly truncated to the same number of bits. In the second scheme, an optimum number of bits were verified for each stage to reduce the hardware without trading off the performance.
a) Uniform Bit Truncation

In Fig. 6.3, various probabilities associated with two signals are compared based on the number of bits used in FFT operation. With the increase of the bits, the detection

Fig. 6.2. Conventional radix-r butterfly architecture.

Fig. 6.3. Receiver performance of conventional FFT with bit truncation.
The probability of the 1st and 2nd signals also increases. The 1st and 2nd signal false alarm is within 1% and the probability of 2nd signal missing is within 20%.

The effect of bit truncation in conventional FFT frequency spectrum is illustrated in Fig. 6.4 where the 2nd signal is 24 dB down of the 1st signal. It can be observed from Fig. 6.4 that as the number of bits used in the FFT computation increases, the spurious signals causing false alarms reduce and the weak second signal is exposed. The minimum number of bits used in the conventional FFT to achieve a dynamic range of 24 db is 11.
b) Optimum Bit Truncation

In this bit truncation scheme, the bits in each stage was truncated to an optimum number of bits to maintain the same performance as in the uniform bit truncation scheme (i.e., 24 dB two-signal IDR, the 2nd signal false alarm is within 1% and signal missing is within 20%). Different combinations of bit truncations for each stage were tested, as shown in (Table 6.1). The optimum bit truncation for stages (1-8) of the 256 point FFT computation achieving the same performance is 4, 5, 6, 7, 8, 9, 9 and 10-b.

<table>
<thead>
<tr>
<th>FFT stage number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>24 dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncation bits</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td>yes</td>
</tr>
<tr>
<td>Optimum case</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Fig. 6.5. Receiver performance of 12 pt kernel FFT with bit truncation.

Fig. 6.6. 12 pt kernel FFT spectrum with bit truncation.
2) **Bit Truncation of 12-point Kernel Function FFT**

A bit truncation study was conducted for the 12-point kernel FFT to study its hardware implementation. In the study, the number of bits used to represent operands in arithmetic operations in the FFT is increased until the receiver performance is achieved. It was observed that at least 20 bits are required to represent the operands.

In Fig. 6.5, various probabilities associated with two signals are compared based on the number of bits used in its FFT operation. The effect of bit truncation in the FFT spectrum of the signals is illustrated in Fig. 6.6. The optimum number of bits used in the 12 pt kernel FFT to achieve a two-signal IDR of 24 db is 20.

![Radix-2 butterfly architecture using LUT.](image-url)
It is shown that in the 256-point conventional FFT the optimum bits allocation has a lower hardware and less computational complexity compared to the uniform bit allocation. Also, the bit-precision required for the conventional FFT implementation using optimal bit truncation is 11 bits which is much less than 20 bits required for the 12-point kernel FFT implementation.

### 6.3 Configurable Look-Up Table (LUT) FFT

Architecture of using look-up tables to implement a radix-2 butterfly is given in Fig. 6.7. The construction of the LUTs for the radix-2 butterfly is illustrated in Fig. 6.8. Based on the level of butterfly, the product of inputs with the sinusoidal function is computed and converted to a binary number represented by a truncated number of bits in LUTs. The LUT is configured for use sequentially and meeting the throughput of parallel butterfly to reduce the hardware at the trade-off of computation time. The performance comparison of conventional FFT, LUT FFT, and configurable LUT FFT will be discussed.

The Radix-2 butterfly architecture as shown in Fig. 6.7 is implemented using two types of multipliers: conventional binary multipliers and LUT multipliers. The comparison of worst case delay and gate count between the two implementations is shown in Table 6.2. The design synthesis and optimization was performed using Cadence PKS in 130 nanometer digital CMOS process.

It is observed that the radix-2 butterflies using the LUT multipliers has an average of 17% reduction in delay and 43% reduction in hardware overhead. Note that the ROM
access and data retrieval time was not added to the delay of conventional binary multipliers. Adding these delay numbers, the delay reduction will be higher.

The gate count and delay estimates before and after LUT configuration for 1024, 512, 256, and 128-point FFT based on the two radix-2 butterflies results (Table 6.2) is presented in Table 6.3. The LUTs are configured to reduce the hardware. For example, in the 256-point FFT, the LUTs are configured four times for multiplication use in butterflies per FFT stage and the total gate count is reduced from 1,165,696 to 278,944 which accounts for a 74.99% hardware reduction. The trade-off of computation time is increasing from 12.32 ns to 47.30 ns. Using the optimum bit truncation on each stage of FFT as shown in Table 6.3, the 1024, 512, 256, and 128 point FFT will achieve a maximum of two-signal IDR of 32, 29, 27.5 and 23 dB respectively with a maximum bit-precision of 12 bits.

![Fig. 6.8. Radix-2 LUT construction.](image-url)
Table 6.2. Gate count and timing for two radix-2 butterflies in different stages of 256-point FFT.

<table>
<thead>
<tr>
<th>256-point FFT Stage No</th>
<th>Radix-2 using conventional binary multipliers</th>
<th>Radix-2 using LUT multipliers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delay * count per Butterfly</td>
<td>Gate count per butterfly</td>
</tr>
<tr>
<td></td>
<td>(ns)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.34</td>
<td>52</td>
</tr>
<tr>
<td>2</td>
<td>0.34</td>
<td>52</td>
</tr>
<tr>
<td>3</td>
<td>1.54</td>
<td>621</td>
</tr>
<tr>
<td>4</td>
<td>1.76</td>
<td>883</td>
</tr>
<tr>
<td>5</td>
<td>1.97</td>
<td>1119</td>
</tr>
<tr>
<td>6</td>
<td>2.07</td>
<td>1561</td>
</tr>
<tr>
<td>7</td>
<td>2.07</td>
<td>1561</td>
</tr>
<tr>
<td>8</td>
<td>2.20</td>
<td>1759</td>
</tr>
</tbody>
</table>

* Delay excluding the ROM access and data retrieval time, and gate count excluding the ROM hardware

The required time for non-configurable LUT 12-b 1024-point FFT processing is 15.78 ns at a clock frequency of 470 MHz. The configurable LUT 12-b 1024-point LUT FFT requires a processing time of 61 ns but the gate count is reduced to 1,392,768 from 5,568,000 of the non-configurable structure.
6.4 Error Analysis of LUT FFT

In this section, an error analysis of the LUT FFT is presented. A 256 point LUT FFT with uniform bit truncation and optimum bit truncation is studied and compared for their performance in the digital receiver (Fig. 6.1). At a constraint of the weak signal missing less than 20% and the false alarm less than 1%, 10,000 simulation runs were taken and each run generated two signals for simulation, i.e., the second signal is 24 dB down of the first signal. The frequency detection error of the second signal for 10,000 simulation runs is presented in Fig. 6.9 in which the detected signals is sorted based on the error from the

Table 6.3. Gate count and delay estimates before and after LUT Configuration for 1024, 512, 256, and 128-point FFT.

<table>
<thead>
<tr>
<th>FFT points</th>
<th>Bit precision per stage</th>
<th>Clock cycles</th>
<th>Gate count</th>
<th>Delay</th>
<th>Gate count</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Max. two-signal IDR</td>
<td>Conf. LUT FFT</td>
<td>Non Conf. LUT FFT</td>
<td>Non Conf. LUT FFT</td>
<td>Conf. LUT FFT</td>
<td>Non Conf. LUT FFT</td>
</tr>
<tr>
<td>1024</td>
<td></td>
<td>32</td>
<td>34</td>
<td>5,568,000</td>
<td>15.78</td>
<td>1,392,768</td>
</tr>
<tr>
<td>1</td>
<td>2 3 4 5 6 7 8 9 10 11 12 12 12 12 12 12 12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>512</td>
<td></td>
<td>29</td>
<td>30</td>
<td>2,330,624</td>
<td>13.57</td>
<td>584,320</td>
</tr>
<tr>
<td>256</td>
<td></td>
<td>27.5</td>
<td>26</td>
<td>1,165,696</td>
<td>12.32</td>
<td>278,944</td>
</tr>
<tr>
<td>128</td>
<td></td>
<td>23</td>
<td>22</td>
<td>522,880</td>
<td>10.58</td>
<td>139,840</td>
</tr>
</tbody>
</table>

*Each LUTs configured four times per stage
actual frequency. The error ranges from a minimum of 1 MHz to a maximum of 10 MHz. Frequency detection errors for three cases (all eight stages of the 256 point LUT FFT are truncated to 9, 11 and 15 bits) are presented.

![Frequency detection error of the second signal](image)

Fig. 6.9. Frequency detection error of the second signal.

The first row in Fig. 6.9 represents the distribution for the case where all the eight stages in the LUT FFT are truncated to 9 bits. As to the error distribution, significant portion of the errors falls in 1~3 MHz and 9~10 MHz errors. Signal detection, signal miss and false alarm of the second signal which is 24 dB down of the first signal is presented in Table 6.4. The 9 bit truncation results in a low second signal detection (43.8%) and a high false alarm (56.2%).
The second and third rows in Fig. 6.9 show the distribution of the other two cases where all eight stages of the 256 point LUT FFT are truncated to 11 and 15 bits. In both cases, the distribution shows a gradual decrease when the frequency detection error increases. Both cases have same second signal detection, miss and false alarm probabilities. It is observed once the optimum bit width requirement is met, there is no significant difference in the achieved results.

The frequency detection error distribution of the second signal for the optimal bit truncation case (in Table 6.1) is given in Fig. 6.10. The distribution also shows a gradual decrease when the frequency detection error increases.

Table 6.4. Signal detection, miss and false alarm of the second signal using LUT FFT.

<table>
<thead>
<tr>
<th>Detection (%)</th>
<th>Miss (%)</th>
<th>False alarm (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncated to 9 bits</td>
<td>43.8</td>
<td>0</td>
</tr>
<tr>
<td>Truncated to 11 bits</td>
<td>82.4</td>
<td>17.6</td>
</tr>
<tr>
<td>Truncated to 15 bits</td>
<td>82.4</td>
<td>17.6</td>
</tr>
</tbody>
</table>

6.5 Summary

In this section, different FFT implementations and bit-truncations were compared in terms of performance in digital wideband receiver and hardware. The conventional FFT implementation required much less bit-precision compared to kernel function FFT.
Radix-2 butterflies were implemented using both look-up tables and conventional binary multipliers, and compared. It was observed that the radix-2 butterflies using the LUT have an average of 17% reduction in delay and 43% reduction in hardware. Practically, the transform length for an FFT system is often more than 64 points. It is desirable that FFT systems with short transform lengths can be expandable and used easily to form new FFT systems with longer transform lengths. The proposed LUT is expandable for any bit truncation for longer lengths butterflies. It can be used as a basic building block for constructing larger FFT. As shown in Table 6.3 the truncated LUT is easily expandable for FFT of variable length (128–1024 point) and variable bit-precision. A detailed frequency error analysis for the optimum bit truncation scheme shows that once the optimum bit width is met, there is no significant difference in the achievable results.

Fig. 6.10. Frequency detection error of the second signal for the optimal bit truncation case
XII. CONCLUSION

Today’s very deep submicron IC technology enables high-performance analog and digital applications to be integrated on a single piece of silicon. Currently, there are growing interests in both theory development and hardware implementation of wideband digital receivers due to advancement in sampling frequency and resolution bits of ADC and high-speed digital signal processors. Desirable characteristics of digital receivers include: 1) wider bandwidth with good frequency resolution, 2) higher instantaneous dynamic range, 3) detection of multiple signals, and 4) faster real-time operation.

Digital spectral estimation is an integral part of a digital receiver that involves determination of power spectrum of the signal. FFT is a popular spectral analysis method widely used in digital receivers. But, if there are two signals with frequencies very close, an FFT operation may only generate one peak containing both the signals. High resolution spectrum estimation algorithms may separate the two signals by generating two peaks instead of one. However, many of them on one hand improve the frequency resolution but on the other hand are difficult to be realized in a real-time system.

Possible approaches to eliminate the two deficiencies are to increase the bit number of the ADC and change the kernel function of the receiver. Increasing the bit number of ADC and the number of bits in the kernel function will require more hardware, which affects the minimum hardware and processing requirements of the digital receiver.
The proposed combination of using a Kaiser window to reduce the spectral leakage by eliminating the discontinuities at the time window edges and using a compensation method to uncover the weak signal extends two-signal IDR of the proposed 1-GHz bandwidth digital receiver to 24 dB. A detailed comparative study of different fixed and adjustable windowing functions was conducted. A compensation technique implemented using a super-resolution algorithm to improve the two-signal IDR by subtracting the estimated peak (strong) signal frequency response and its spurs from the actual received frequency response to expose the weak signal was presented. Both the Kaiser window and the super-resolution algorithm were implemented and performed in a real-time system. An exhaustive study of configurations of ADC, FFT, window function, and compensation for maximum achievable two-signal IDR was presented.

7.1 Research contributions

The proposed 1 GHz bandwidth digital receiver was implemented using the IBM 130 nm 8SFG CMOS standard cell library. The design was coded in Verilog and its function verified using DAI Signalscan. Cadence BuildGates was used to perform logical synthesis and optimization. Cadence PKS was used to perform placement and routing. The chip design was broken down into six major subsystems with total of 2,205,538 transistors and has a die size of approximately 1.98 mm x 1.98 mm. The hardware implemented for the Kaiser Window and the compensation can process the two signals in 40 ns, less than the time allotted for the data processing (100 ns).

Multiple signal detection in a noisy signal environment is often a challenge due to restriction of limited hardware and timing constraint in real-time implementation. A
configurable digital receiver with multiple signal detection capability was presented. The proposed design employs hardware reuse and detects up to five signals with maximum two-signal IDR of 17 dB.

Hardware and performance of wideband digital receivers using conventional FFT and kernel function FFT were studied and compared. To achieve the same two-signal IDR, the conventional FFT requires much less bit-precision at all stages compared to the kernel function FFT, but its implementation faces a challenge due to too many binary multipliers needed. In this research, two radix-2 butterflies at different stages of 256-point FFT implemented by look-up tables (LUTs) and by conventional binary multipliers was compared. It was observed that the radix-2 butterflies using the LUTs have an average of 17% reduction in delay and 43% reduction in hardware. The LUTs are configured to reduce the hardware. For example, in a 256-point FFT the LUTs are configured four times for multiplication use in butterflies per FFT stage and the total gate count is reduced from 1,165,696 to 278,944 which accounts for a 74.99% hardware reduction. The computation time is increased from 12.32 ns to 47.30 ns. The configurable LUT FFT processor with short transform lengths can be expandable so that they can be used easily to form new FFT processors with longer transform lengths. The performance comparison of conventional FFT, LUT FFT, and configurable LUT FFT for wideband digital receiver application was discussed.

In summary, this dissertation includes four substantive contributions:

- A Kaiser window function was analyzed and selected to perform data windowing on ADC digitized data in the wideband receive. A novel hardware
implementation of the Kaiser window to save hardware and reduce delay was presented.

- Design and implementation of the compensation method using super-resolution algorithm was presented. Several high-resolution algorithms have been proposed in the past, but many of them are not suitable for real-time implementation.

- Design and implementation of a configurable receiver in which an efficient multiple signals detection scheme employing hardware reuse by effective configuration was presented.

- A configurable and expandable FFT processor design was presented. A LUT methodology was developed and demonstrated on variable length (128–1024 point), variable bit-precision (6-12 b) FFT, with uniform bit truncation and optimum bit truncation for wideband digital receiver in radar applications.

7.2 Future Work

Although several researches have been conducted in wideband digital receivers to date, there are still many issues to be resolved. For example, new high-resolution algorithms are in demand to achieve a higher two-signal IDR and a smaller false alarm that are desirable characteristics of future modern wideband receivers. The achievable bandwidth of the receiver is technology dependent. With ADC and DSP technology advancing at a fast pace, much higher bandwidth is to be realized. This would drastically decrease the search time for the frequency of interest and improve the response time of the receiver.
Computationally efficient hardware implementation and methodology needs to be further explored to improve upon the existing bandwidth.

With the advance in IC technology and recent explosive growth in the wireless telecommunication, there is a growing trend for RF integration in CMOS. Although several RF front-end systems have been proposed in the past, developing a highly CMOS integrated RF receiver is an extremely challenging task and requires innovative new circuit architecture and technology, and advanced design skill. The challenge is to integrate the RF/mixed-signal circuits like low noise amplifiers, mixers, filters, ADCs along with the digital signal processor on a single chip. Currently wireless telecommunication devices make use of RF receiver chips based on relatively expensive process like SiGe or BiCMOS. However, CMOS process with its promising low power and low cost benefits is predicted to leverage higher performance scaling of Moore's Law which not offered by SiGe or BICMOS process.
Bibliography


