Optimized Soft-Core Processor Architecture for Noise Jamming

By

Madodana Mfana

Dissertation submitted in partial fulfilment of the requirements for the degree

Master of Electrical & Electronic Engineering

at the Faculty of Engineering and Built Environment

University of Johannesburg

December 2019

Supervisors: *Dr Ali Nabil Hasan*

Dr Ahmed Ali

Declaration

I, **Madodana Mfana** declare that:

i. The research reported in this dissertation, except where otherwise indicated, is my original work.

ii. This dissertation has not been submitted for any degree or examination at any other university.

iii. This dissertation does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

iv. This dissertation does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

   a. Their words have been rewritten but the general information attributed them has been referenced;

   b. Where their exact words have been used, their writing has been placed inside quotation marks, and referenced.

v. This dissertation does not contain text, graphics or tables copied and pasted from the internet, unless specifically acknowledged, and the source being detailed in the dissertation and in the References section

Signed: ……

Name: Madodana Mfana

Acknowledgements

The past year has been long and exhausting in some regards and challenging with moments of excitement. I would not have made it to this point without the people who stood by me and rendered much-needed support. I extend my sincere gratitude to my supervisors, Dr Ali Hasan and Dr Ahmed Ali. Their guidance and assistance helped me eventually finish this dissertation within the targeted timelines. I also thank the team at Sovereignty Systems (Pty) Ltd that includes Mr Yanga Tekane with assistance on operation of experimental equipment thank you all.

To my wife Ipeleng Mfana – I'm thankful for all the support you continue to give me emotionally and spiritually, I also want to thank her for understanding and patience when I took time away that should have been spent on family. A big "thank you" to my children Aqhamile and Qhamani for their unwavering love and support. I thank my lovely mother-in-law Rebecca Monnahela for her support and prayers.

Publications

DETAILS OF CONTRIBUTION TO PUBLICATIONS that form part of this dissertation, or include research presented in it, are as follows:

Publication 1:

Madodana Mfana, Ali N Hasan, and Ahmed Ali, “Odd/Even Order Sampling Soft-Core Architecture towards Mixed Signals Fourth Industrial Revolution(4IR) Applications”, *Energies MDPI Journal.* Volume 12, Issue 23, 4567, 29 November 2019

Author contributions: M Mfana developed the mathematical model, simulation model, assisted with conducting experiments, analysed simulation and experimental results, and wrote the paper. The other authors assisted with reviewing the paper.

Abstract

Noise jamming is a traditional electronic counter measure (ECM) that existed since the establishment of electronic warfare (EW). Traditional noise jamming techniques have been shown to be failing when interacting with intelligent Radar systems such as pulse Doppler radar. Hence there is a need to introduce new noise jamming techniques with digital architecture that will provide improved performance against smart pulse Doppler radar. The work is undertaken to investigate the feasibility of digitizing noise jamming. It focuses on analog-to-digital conversion optimization towards noise jamming architecture, as a result digitization will allow for an opportunity for adaptation of intelligent processing that previously didn't exist.

In this dissertation, certain contributions to the field of noise jamming were made by introducing state of the art odd/even order sampling architecture by proving four case studies. Case study 1 experimentally investigates sample frequency behaviour. Case study 2 uses simulation to investigate step-size and dynamic range behaviour. Case study 3 uses FPGA implementation and SNR to investigate quantization error behaviour. Case study 3 also uses SNR to investigate superiority of proposed odd/even order sampling. Lastly case study 4 uses field measurements, FPGA implementation and SNR to investigate practical implementation of digitized noise jamming.

The main contribution is concerned with an architecture that digitizes, reduces sample frequency, optimizes digital resource utilization while reducing noise jamming signal-to-noise ratio. The approach evaluates and empirically compares three sampling techniques from lecture Mod-Δ, Mod-Δ (Gaussian) and Mod-Δ (Sinusoidal) with proposed novel odd/even order sampling. Sampling techniques are evaluated in terms of quantization error, mean square error and signal-to-noise ratio.

It was found that the proposed novel odd/even order sampling achieved most case SNR performance of 6 dB in comparison to 18 dB for Mod-Δ. Sampling frequency findings indicated that the proposed novel odd/even order sampling had achieved sampling frequency of 2 kHz in comparison to 8 kHz from traditional 1st order sigma-delta. Dynamic range findings indicated that the proposed odd/even order sampling achieved a dynamic range of 1.088 volts/ms in comparison to 1.185 volts/ms from traditional 1st order sigma-delta. Findings have indicated that the proposed odd/even order sampling has superior SNR and sampling frequency

performances, while the dynamic range is reduced by 8%. It was also found that signal-to-noise ratio results for the proposed noise jamming outperforms gaussian noise jamming by at least 4 dB. Efficiency of traditional noise jamming as characterized by simple architecture and operation, was conducted. Noise jamming has not been studied to this extent until now.

Keywords: Soft-Core Architecture; Odd Order Sampling; Even Order Sampling; Analog to Digital Converter (ADC).

Table of Contents

Table of Figures

List of Tables

Nomenclature

1.	RAM	Random Access Memory
2.	FPGA	Field Programmable Gate Array
3.	LUT	Look Up Table
4.	DSP	Digital Signal Processing
5.	ILA	Integrated Logic Analyzer
6.	RMS	Root Mean Square
7.	ARMS	Average Root Mean Square
8.	PDF	Probability Distribution Function
9.	CDF	Cumulative Distribution Function
10.	PSD	Power Spectral Density
11.	AR-PSD	Autoregressive Power Spectral Density
12.	ADC	Analog-to-Digital Converter
13.	SNR	Signal-to-Noise Ratio
14.	SQNR	Signal-to-Quantization-Noise Ratio
15.	R	Bit Rate
16.	D	Distortion
17.	dB	Decibels
18.	dBm	Decibel-milliwatts
19.	dBFS	Decibels Relative to Full Scale
20.	dBTP	Decibels True Peak
21.	MSE	Mean Square Error
22.	DR	Dynamic Range
23.	Mod-Δ	Mod-Delta
24.	REW	Radar & Electronic Warfare
25.	SAR	Synthetic Aperture Radar
26.	ISAR	Inverse Synthetic Aperture Radar
27.	EW	Electronic Warfare
28.	ECM	Electronic Counter Measure
29.	ECCM	Electronic Counter-Counter Measure
30.	ESM	Electronic Support Measure
31.	RGPO	Range Gate Pull Off
32.	VGPO	Velocity Gate Pull Off
33.	AGPO	Angle Gate Pull Off

34.	RSP	Radar Signal Processor
35.	DRFM	Digital Radio Frequency Memory
36.	FMCW	Frequency Modulated Continuous Wave
37.	AI	Artificial Intelligence
38.	ML	Machine Learning
39.	ANN	Artificial Neural Network
40.	BPNN	Back-Propagation Neural Network
41.	FNN	Fuzzy Neural Network
42.	GA-NN	Generic Neural Network
43.	GA	Generic Algorithm
44.	Hz	Hertz
45.	kHz	Kilohertz
46.	MHz	Megahertz
47.	GHz	Gigahertz
48.	I	Current
49.	V	Voltage
50.	DoE	Design of Experiment

Chapter 1: Introduction

3.1.1.Electronic Warfare and Electronic Counter Measures Overview

Electronic Warfare (EW) is an umbrella field of practice for a set of electronic attack and electronic intelligence systems used as target information protection against Radar. Electronic attack systems are usually being referred to as Electronic Counter Measures (ECM) and electronic intelligence systems usually referred to as Electronic Support Measures (ESM). ECM are divided to passive jamming and active jamming, and well-known passive jamming techniques are noise and chaff jamming. Well known active jamming techniques are Range Gate Pull Off (RGPO), Velocity Gate Pull Off (VGPO) and Angle Gate Pull Off (AGPO).

Electronic Counter Measure (ECM) techniques have always been an information technology battle ever since their inception during World War 2 [1]. The outcomes of various information exploitation experiments are vital in protecting both manned and unmanned airborne platforms [1]. The development of jamming waveforms and ECM techniques are based upon data exploitation and circuit analysis of various threats hardware and firmware [1]. Almost all modern tracking Radars use range gates in the digital signal processing (DSP) circuitry to compute range to target [1]. The range gate is an electronic switch which opens after some time delay t_d following the pulse transmission path and the gates open for a few milliseconds [1]. If a return pulse above the automatic gain control (AGC) is detected while the switch is open, then the range to target is given by:

$$R_T = t_d \frac{c}{2}$$

Where

$$R_T = Range\ in\ Km\ to\ the\ target$$

$$t_d = Time\ delay\ of\ the\ range\ gate$$

$$c = speed\ of\ light$$

A technique called Range Gate Pull Off (RGPO) is used by a stand-off jammer to try and confuse the enemy Radar into thinking the target is at different range [1]. When the jammer detects an enemy's radar signal, the deception jammer then memorizes and retransmits the

signal, but at a stronger signal-to-noise ratio (SNR) than the original radar return. This is depicted in Figure 1.

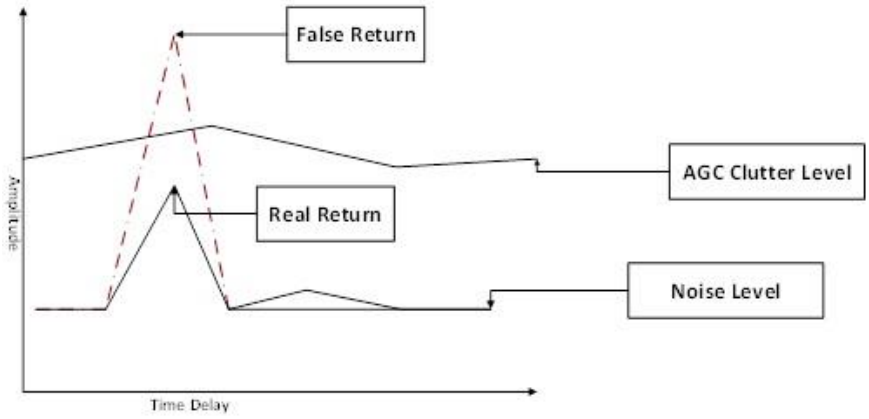


Figure 1. *This portrays general form of jamming, showing parameters at play during the attack*

The jammer also has the advantage of one-way transmission which required half the power of the enemy radar's two-way radar transmission. Once the enemy radar is fooled by this technique, then all the jammer must do is "walk" the radar return resulting in the enemy radar tracking a false target [1].

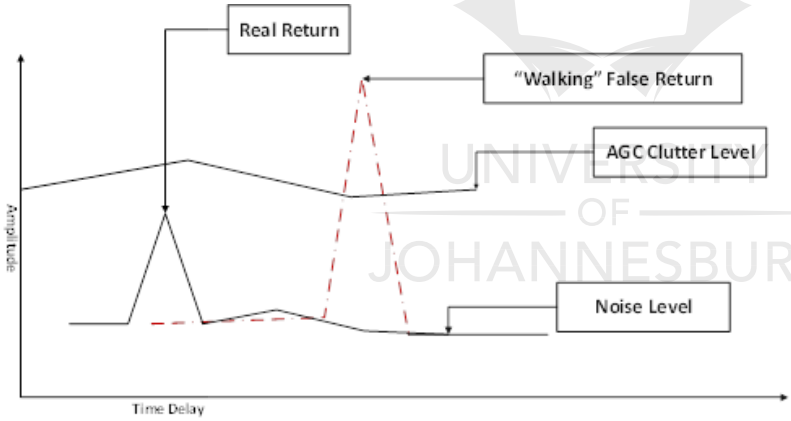


Figure 2. *This portrays general form of RGPO, showing parameters at play during a RGPO attack*

RGPO is less effective against modern radars whose fast microprocessors can run many DSP algorithms [1]. Nevertheless, RGPO is still in use. Its simplicity makes it an ideal test bed to prove that one can use a GA to derive electronic counter-counter measure (ECCM) techniques given various other parameters [1]. If successful, then GAs will be tried on more sophisticated ECCM techniques [1]. An ECM skin cover pulse is frequently synthesized to be coincident, or as closely coincident as possible, to the true skin return [1]. The primary reason for its use is that it is difficult to control the J/S seen by the radar: if the jamming signal pulses are uniform

and substantially higher power than the true skin return, then an amplitude discriminant may be used by the radar to help identify the skin return [1].

Note that this definition of a cover pulse differs from the one used in the 1995 Benchmark Model, where the cover pulse was a non-coherent pulse that did not compress in the radar receiver [2]. The cover pulse specified here is a replica of the transmitted pulse and will produce a coherent detection in the same detection cell as the skin return. A pulse diversity-based jammer cancellation processor is designed in this literature to cancel the deception electronic counter measures (ECM) signals repeated by a digital radio frequency memory (DRFM) for radar systems [3]. To remove the conventional stationary hypotheses and the rigorous application conditions, an orthogonal transmission block code set is achieved in one pulse repetition interval by exploiting the orthogonal frequency division multiplexing (OFDM) [3]. The processor does not require parameter estimation and compensation across a period of a whole block code [3].

2.1.1. Noise Jamming Overview

Shixiang and Xizhang [4] describe noise jamming as the use of randomicity of signal to protect target information. Li, Tai. Wang and Yuan[5] introduce a passive jamming method such as chaff jamming which is commonly used to jam Radar. In general terms chaff jamming doesn't need a special electronic system but uses the reflective properties of metallic pieces to deflect radar and confuse the enemy, the target echo is protected from radar by a cloud of tinny metals released by the target [5]. Noise jamming is a form of passive jamming having two major advantages to EW designs when compared to active jamming techniques, such advantages are:

- Simplified electronic system architecture
- Minimum cost expenditure for EW manufacturing

Xun, Cui and Hao[6] claim that noise frequency modulation (FM) is the main form of noise jamming. Xun, Cui and Hao [6] also concluded this claim after conducting an empirical study on parameters of noise FM selection. Literatures [7, 8, 9, 10, 11] present a combination of studies leading to a single goal of diffusing Synthetic Aperture Radar (SAR) target image information. Hong-xu, Yi-yu, Wen-li and Zhi-tao [7] use a modified time delay jamming to introduce point and line image error to SAR target image. It can be noted in their results that line errors are enhanced in the jammed image when compared to the noise jammed one. It should be also noted that the general quality of the SAR image becomes horrible after being

jammed, but only straight lines which are modified and made thicker with time delay methodology are presented in [7].

As a matter of background, SAR is a coherent airborne or space borne side positioned radar system. In operation SAR is configured in a similar way to that of phased array, in contrast to many parallel antenna elements in phased arrays, SAR uses one antenna in time-multiplex. H. Hong-xu, Yi-yu, Wen-li and Zhi-tao [8] also conducted a related study in the frequency domain. Hong-xu, Yi-yu, Wen-li and Zhi-tao [8] developed a new algorithm named modified shift frequency jamming which introduce noise to the SAR image, the resulting in a blurry image. It is important to note that studies in [7] and [8] are related in a sense that a time delay in the time domain is a frequency shift in the frequency domain, but the application of these two methods achieves totally unrelated results. Lv [9] explore a simulation study for the behaviour of noise convolution jamming for attacking SAR images, the simulation results are comparable with those obtained in [8]. Literatures [12, 13, 14] present a combination of studies leading to a single goal of diffusing ISAR (Inverse Synthetic Aperture Radar) target images information.

Diffusing of pulse Doppler radar is an important research topic in EW engineering practice. Although several studies have been conducted on EW, no satisfactory methods combining pulse Doppler processing and dual-port memory mapping with noise jamming have been developed to outperform pulse Doppler radar. This lack of knowledge has led to the development of active jamming techniques with computational challenges resulting in an expensive electronics system cost.

2.1.2. Soft-Core Architecture

As field programmable gate array (FPGA) technology has steadily improved, FPGAs have become viable alternatives to other technology implementations for high-speed classes of digital signal processing (DSP) applications. Radar front-end signal processing, an application formerly dominated by custom very large-scale integration (VLSI) chips, may now be a prime candidate for migration to FPGA technology [15]. As this literature demonstrates, current FPGA devices have the power and capacity to implement a FIR filter with the performance and specifications of an existing, in-system, front-end signal processing custom VLSI chip. A 512-tap, 18-bit FIR filter was built that could achieve sample rates of 7 MHz (with a clock rate of 117 MHz) using Xilinx Virtex FPGA technology and was demonstrated through simulation and hardware implementation.

In literature a Xilinx Virtex 5 FPGA platform-based signal processing algorithm has been developed and experimentally verified for use in a MEMS based tri-mode 77GHz FMCW automotive radar to determine range and velocity of targets near a host vehicle. The Xilinx Virtex 5 FPGA based signal processing and control algorithm dynamically reconfigures a MEMS based FMCW radar to provide a short, medium, and long-range coverage using the same hardware. The MEMS radar comprises of MEMS SP3T RF switches, micro-fabricated Rotman lens and a micro strip antenna embedded with MEMS SPST switches, in additional to other microelectronic components.

Current radar signal processor architectures lack either performance or flexibility in terms of ease of modification and large design time overheads. Combinations of processors and FPGA are typically hard-wired together into a precisely timed and pipelined solution to achieve a desired level of functionality and performance. Such a fixed processing solution is clearly not feasible for new algorithm evaluation or quick changes during field tests. A more flexible solution based on high-performance soft-core processing is proposed. To develop such a processing architecture, data and signal-flow characteristics of common radar signal processing algorithms are analysed. The proposed architecture achieves high data throughput while providing the flexibility that a software-programmable device offers. The end user can thus write custom radar algorithms in software rather than going through a long and complex HDL-based design. The simplicity of this architecture enables high clock frequencies, deterministic response times, and makes it easy to understand.

In modern radar systems, the architecture of the radar signal processor (RSP) is one of the most important design choices. The amount of useful information that can be extracted from the radar echoes is highly dependent on the computational performance that the RSP can deliver. Current RSPs lack either performance or flexibility in terms of ease of modification and large design time overheads. Combinations of processors and field-programmable gate arrays (FPGAs) are typically hard-wired together into a precisely timed and pipelined solution to achieve a desired level of functionality and performance. To address this gap between performance and flexibility, a custom processor architecture is proposed, the proposed architecture performance has been compared to other manufactures.

3.1.2.Research Objectives

Noise jamming is the simplest traditional type of electronic counter measure (ECM) that was used to traditionally attack Radar systems. Improvement in computational intelligence has led to development of modern pulse Doppler radar that outperforms traditional noise jamming. There is a need to upgrade the architecture of noise jamming to a level it can compete with pulse Doppler radar while retaining its simple functionality, and use simple soft-core processing to introduce intelligence to noise jamming.

The work that was undertaken introduced state of the art odd/even order analog-to-digital sampling, soft-core architecture from existing building blocks, and signal-to-noise ratio study of the proposed and existing sampling schemes. The primary major objective of this dissertation is to illustrate how odd/even order analog-to-digital sampling can play a critical role in optimizing and simplifying the received Radar signal in noise jamming architecture. The secondary major objective is to a new ordered analog-to-digital sampling scheme with efficient step-size, dynamic range and sample frequency. The tertiary major objective is a study of the proposed sampling scheme in comparison to existing sampling schemes. If successful, ordered analog-to-digital sampling might lead to optimized electronic warfare (EW) architecture with efficient computational specifications.

3.1.3.Synopsis of the Dissertation

Chapter 1 introduces the study. Electronic warfare is introduced and the general problem of electronic attack architecture is also presented. A short summary of noise jamming and soft-core architecture using support from evidence from literature, is outlined

Chapter 2 introduces certain aspects of widely used digital building blocks, such as I/Q demodulator, channel equalization, analog-to-digital sampling, amplitude detector, phase detector and frequency detector. Widely used methods and algorithms, such as Kalman Filter, Peak Detection and Modified-Costas are described.

Chapter 3 derives the mathematical model from digital building blocks, factors advanced noise jamming techniques such as modified shift frequency, and convolution noise jamming. It develops requirements for simulation and implementation.

Chapter 4 presents case study 1 with an experimental setup consisting of signal synthesizer producing quadrature signal, quadrature demodulator, oscilloscope and spectral analyser.

Quadrature signal design was conducted on both the signal synthesizer and the spectral analyser.

Chapter 5 presents case study 2 with an experimental setup consisting of Matlab Simulink with a developed delta-sigma model. The delta-sigma model was modified to cater for sparate odd and even order sampling model. Simulink tools such as the signal generator and an oscilloscope were used for signal analysis.

Chapter 6 presents case study 3 with an experimental setup consisting of a MiniZed FPGA development board and a Vivado design suite. The vivado measurement tool such as integrated logic analyser was used for signal analysis.

Chapter 7 presents case study 4 with an experimental setup consisting of a MiniZed development board and a Vivado design suite like case study 3. The vivado measurement tool is also used like case study 3.

Chapter 8 presents a summary of results of the study with recommendations and suggestions.

Chapter 2: Current Soft-Core Architecture Techniques

2.2. Introduction

This chapter starts by defining the term digitization in the context of this research, and further defines building blocks. Building blocks were largely gathered from literature and these include the I/Q demodulator, channel equalization, digital sampling, phase and frequency detection. Finally, an overview of available sampling schemes and techniques, points such as material and methodologies used and developed, were emphasized.

2.3. Digitization Definition

Digitization is the computational process preceding advanced digital techniques such as digital signal processing, Radar signal processing and machine learning. The heart of the computational process is a sampling approximation of the analogue signal through level-based quantization techniques. Digitization is at the centre of 4IR with previously analogue system being digitized through the analog-to-digital converter. For 4IR applications such as 5G Cellular Networks Technology and Cognitive Electronic Warfare (EW) at some point interface to digital through analog-to-digital converter. Efficient use of digital resources such as memory largely depends on the sample design of the analog-to-digital converter. Existing even order sampling has been found to perform better than traditional sampling techniques.

2.3.1. Analog-to-Digital Sampling and Soft-Core Architecture Building Blocks

2.3.1.1. Analog-to-Digital Sampling and Soft-Core Architecture Building Blocks

Prototype Integrated Circuit (IC) with a goal to achieve low power computation, a final design with specification 190MHz I/Q demodulator with amplitude gain mismatch of 0.1dB, phase mismatch of 1.25º and non-specified frequency mismatch was achieved by the Marzuki, Shakaff and Sauli [16]. Resource utilization specified by the author are 2 mixers, local oscillator (LO) divider, negative resistance circuit, buffer amplifiers and bias circuits [16]. Slyusar and Serduk [17] look at cascading two I/Q demodulators and obtaining peak-frequency characteristic, this design approach allowed for noise immunity on the combined I/Q demodulators. A novel I/Q demodulator to extract amplitude and phase for Continuous Wave

(CW) ultrasonic, the design makes use of phase shift selection [18]. Design implementation and testing were done on AD8333 and AD8332 with state-of-the-art $0.8\,{}^{nV}/_{\sqrt{Hz}}$, 19dB plus differential amplifier with +5V single rail supply.

A digital I/Q demodulator that eliminates digital filters is used to overcome problems with amplitude and phase mismatch in I and Q channels [19] test results showing quantization noise were presented showing the performance of a digital I/Q demodulator. Literature [20] introduces the idea of ultrasound complex signal generation using CORDIC algorithm for transmission and mixing, the system design is for measuring and analysis of forced sinusoidal vibration with frequencies ranging from 100 to 400 KHz.

PCB design following FPGA design process with Xilinx XC3SD3400A Spartan 3A DSP falling under 90 nm silicon technology as the heart of the system was developed [20]. Implementation architecture focused on optimizing resource utilization on the FPGA as they are available at limited quantities [20]. A multi-channel digital I/Q demodulator is proposed in the literature but evidence of its experimental result and performance parameters have not been confirmed. Overall system optimization architecture was presented, and logical architecture design was presented but results are also non-existent [20]. Literature [20] present a laboratory experimental setup but do not explain to us how the multi-channel digital I/Q demodulator is implemented and tested so that we can use it as a benchmark in our proposed study.

An odd number Analogue to Digital Converter is selected as input to the digital I/Q demodulator to help relieve special voltage stress in the receiving segment of a radar systems [21]. Literature [21] presents a new class of I/Q demodulators with odd order derived from the even order I/Q demodulator which is characterized by linear phase-frequency relation for wide band signals. The odd order I/Q demodulator presented in [21] solves the signal syntheses problem. The mathematical representation of the odd order I/Q demodulator are given in equations 1 and 2.

$$\mathrm{I} = u_1 - 3u_3 \; where \; u_i \; is \; the \; ADC \; samples \tag{1}$$

$$\mathrm{Q} = -(3u_2 - u_4) \; where \; u_i \; is \; the \; ADC \; samples \tag{2}$$

HMIC integrated circuit (IC) with structural components of silicon and glass for RFID design, the HMIC IC is used to create a 90 degree shift I/Q Modulator/Demodulator for binary coding of RFID tags [22]. CMOS 90 nm silicon design using transformer coupling is presented in

literature [23], the I/Q demodulator focus is on a mixer where there is a replacement of cascaded transistors with transformer coupling. Literature [21, 23] present two interesting design approaches to the I/Q demodulator. Literature [21] proposes a selection of odd ADC samples and eliminating even samples. Eliminating odd or even samples means that half the wave data is used without compromising complex signal parameters such as amplitude and phase, elimination of half of the wave data could improve processing requirements such as signal mixing, accumulation, memory and buffering. Literature [23] looks at something specific such as efficient complex signal mixing in the I/Q demodulator, they improve complex signal mixing design by replacing traditional cascaded transistors design with transformer coupling.

Literature [24] presents an IC design for 65nm CMOS fabrication of 60 GHz I/Q modulator/demodulator for short range wireless commutation with a characteristic range of 10 to 100m. A 60-GHz Direct Conversion I/Q Modulator/Demodulator CMOS architecture is presented with differential signal levels for both I and Q [24].

Table 1. *Performance of different fabrication technologies*

Literature	Bandwidth	Technology	Ampl. Imbalance	Phase Imbalance
[16]	190 MHz	SiGe 0.35µm BiCMOS	0.1dB	1.25º
[24]	800 MHz	65nm CMOS	0.7dB	4.5º
[25]	100 MHz	90nm CMOS	0.1dB	5º
[26]	2 GHz	0.35µm CMOS	-	2.8º

An I/Q modulator/demodulator with 1.6 Gb/s data rate for 16-QAM 90nm CMOS fabrication design is presented [25]; the focus is on an I/Q demodulator design for high data rates on the E-band with a sub-harmonic down-conversion resistive mixer. A complex phase shifter is designed to draw the I and Q channels from the complex signal for operation at 2GHz, the design was fabricated on 0.35µm CMOS with phase imbalance of 2.8º [26].

[27] aims to design a front-end DSP I/Q modulator / demodulator for the IEEE 802.11a WLAN test-bed, resource utilization for both modulator and demodulator is 120K FPGA gates is 2% usage of Xilinx FPGA XC2V-6000. [28] introduces the use of tandem scheme of two stage Signal Processing for design of an I/Q demodulator, a tandem scheme has a serial number reference from Analogue to Digital Converter (ADC), the quadrature voltage components and weighted signal processing array. [29] improves the performance of an I/Q demodulator by using band pass and low pass filters while considering the approach of constructing a multilayer I/Q demodulator.

Kalman filter with least mean square (LMS) is used as an adaptive channel equalizer, while the LMS and recursive least square (RLS) are used for predicting channel coefficients [30]. Two new channel equalization algorithms are derived from Kalman filter predictive algorithm optimized for inter-symbol interference (ISI), these algorithms are named K-LMS and K-RLS which were studied through a simulation [30]. Phase gradient techniques were used to estimate the aberrant phase from resolution cell in the range-azimuth, phase correction was designed [31].

Authors in literature [32] present an IC design that implements a compensation technique for amplitude and phase imbalance, the heart of the compensation technique is a self-aligned tuner optimized for digital modulation. The self-aligned tuner is a mixed signal IC that automatically corrects the amplitude and phase and a simulation model was used to test the performance of the tuner. The simulation results showed an amplitude correction from 1dB to 0.052dB and phase correction of 5º to 0.57º [32].

Lu [33] evaluates impacts of amplitude and phase imbalances on quadrature phase shift key modulation, channel model simulation is used as a tool to evaluate the impacts some of the channel activities which were modelled were Inter-Channel Interference and I/Q cross-talk power ratio. Simulation model results in [33] showed that for 1dB amplitude and 5º imbalance the Bit Error Rate (BER) remained below 0.5dB.

Introduction Body Area Networks (BAM) where performance is traded over lower power consumption [34], a simulation model that defines minimally required receiver specifications the model could also be used in gaining insight into demodulator power behaviour. Literature [35] looks at analogue signal imbalances between the In-phase and Quadrature branches of the I/Q modulator/demodulator and the authors develop a compensation technique for amplitude and phase. The compensation technique focusses on resolving impairments due to mixer and

filter imbalances. Carrier frequency offset estimation for I/Q imbalance is studied in [36], leaner least mean square estimation provided high accuracy estimation with lower complexity when it comes to implementation; on the other hand, the Kalman filter demonstrated good channel time variations tracking.

2.3.1.2. I/Q Digital Sampling

In Electronic Warfare (EW) mixed signal processing methods have become normal processing techniques where analogue Radar is sampled to digital using ADC. Pulse Doppler radar techniques such as Frequency Modulated Continuous Wave (FMCW) and Linear Frequency Modulation (FM) with variant frequency introduces the problem of digital formation of I/Q signal in the EW segment [21].

During digital formation process several factors such as sample frequency, the number of bits produced and mean squared error distortion between input and digitally reconstructed signal are evident. A novel even order sampling I/Q demodulator date back as far as [37], and was later followed by the development of odd order sampling I/Q demodulator as recently as 2015 [21]. The odd order I/Q demodulator which is derived from an even order I/Q demodulator, is characterized by linear phase/frequency relation. Such a signal attributes help relieve complexities in digital formation such as samples produced per second and distortion error of for wide bandwidth I/Q signals [21, 37].

Moshkin, Nikolaev and Nikitin [38] study the complexity of the dynamic range related to the number of bits produced through quantization resolution while exploring phase. Several measurements error performances with different resolutions were conducted phase, 1-bit quantization showed the worst phase error performance. To reduce the number of bits produced by the ADC, dynamic range quantization encoder digital architecture design is presented in [39]. Number of bits produced introduce a sample storage capacity complexity, sample storage capacity is addressed by the design of sample and hold circuit with the purpose to extend the input signal range which allows for higher input voltages to be tolerated [40], [41] is also addressing similar issue but using the Multisim simulation tool.

Advances in solutions that address digital formation have led to the development of design and development of CMOS technology prototypes [42, 43, 44, 45, 46, 47]. To optimize the dynamic range Ordentlich, Tabak, Hanumolu, Singer, and Wornell [48] introduces the idea of modulo ADC which reduces the dynamic range of the input signal, this technique allows 5 volts signal to be processed as 2.5 volts for a modulo 2 and processed as 1.5 volts for a modulo

4 implementations. A second order delta-sigma implementation is introduced in [49]; the block architecture is like that of first order delta-sigma design with a difference of a second order integrator before the quantization step. A high-performance DAC approximated ADC architecture for Synthetic Aperture Radar (SAR), the only architecture advance is the switch mode which allows the switch between positive and negative quantitation [50, 51].

To address dynamic range quantization inefficiencies a low bit error encoder is designed and tested using CMOS technology [52]. The different approach to the design that addresses dynamic range resolution, [53] aims to increase the dynamic range by using the advantage of capacitance charging capabilities. The capacitance acts as a sample and hold stage after the quantizer thus allowing them to maintain high the resolution digital signal. [54] achieve the same designs but at low power. Different digital conversion technologies were evaluated towards the design of a PID controller [55]. For digital control the ADC stage is very important and maintaining a high-quality control feedback signal in most cases is vital. Just like in [53] and [54] in SAR imagery high resolution signal as always, the target and [56] is trying to achieve the same objective with 1 bit/multi-level quantization stage. A 3-stage neural network nonlinear ADC was designed and implemented in Matlab Simulink in [57], 16-bit quantization high resolution was demonstrated in the simulation results. A design like [57] is presented in [58] but it is not clear whether it's linear or non-linear. In EW processing the ADC stage is usually followed by enveloped detector stage [59] a mathematical model coherent envelope detector for ambient backscatter is presented. Dynamic range quantization bit resolution complexities are corrected by using the envelope detector stage [60]. Glow plasma is used to detect electromagnetic wave, full digital RF chain is also implemented [61].

2.3.1.3. Envelope Detector

After the envelope detector a phase or frequency detector stage usually follows. Phase detector with feed forward noise cancellation architecture, which uses sub-sampling for phase noise detection [62] . To implement an XOR phase detector an FPGA architecture was design and implemented in [63]. Digital phase locked loop for Internet-of-Things application is presented [64]. Similar to [64] a sampling phase detector with ultra-low phase noise is design for the microwave synthesizer's applications [65]. Simple Bang Band phase detector with 1-bit third-order single feedback loop delta-sigma ADC pre-processing stage for FMCW wave synthesis [66]. Similar to [65] a pulse phase-frequency detector for high-speed frequency synthesizers

applications [67]. A capacitive power sensor is used to design a large dynamic range phase detector [68].

The authors in this article [69] designed an empirical estimation algorithm that takes advantage of prominent amplitude peaks and uses interpolation (iterative search of maxima) to extract the envelope from the peaks and the algorithm was tested on audio waveforms and the results were positive [69]. It is commonly agreed that the envelope is a signal that varies slowly and pass the prominent peaks of data smoothly and in classical perspectives. Fourier spectrum is a critical indicator for evaluating the varying rate of an envelope [69]. The most intuitionistic view on the envelope estimation problem is to draw a curve that outlines the signal contour and passes through the main peaks [69]. A peak detection algorithm combined with moving window was proposed in [70], the proposed empirical algorithm then uses a low pass filter to smooth the quantization noise produced while generating envelope. In simple terms the algorithm first eliminates the negative part of the receives digital signal by taking an absolute value of all the individual samples, the proposed empirical algorithm then defines a processing window and fit enough samples [70]. From each processing window a maximum sample is taken and marked as peak, and all sample elements in the window are replaced with the peak [70], the final processing stage is the filtering which smoothens the envelope.

Autoregressive power density (AR-PSD) was used for homomorphic filtering to achieve Frequency Alterable Homomorphic Filtering (FAHF) proposed in [71]. The proposed FAHF is applied to multichannel cardiovascular field, the performance of the proposed algorithm is compared to Normalized Shannon Energy (NASE) and the Hilbert Transform (HT) algorithm, and the results showed that FAHF produced smoother envelope curves [71]. Experimental setup proposed in [71] consisted of traditional stethoscopes, digital stethoscopes, computer, audio amplifier, BIOPAC physiological signal recorder and four auscultation chest pieces. Butterworth band pass filter and threshold shrinkage denoising which included normalizing the signal by setting the variance of the signal to 1.0, these processing efforts were considered as a pre-processing stage [71]. Authors in [71] also discovered during the experimental stage that the proposed FAHF algorithm can adaptively denoise multiplicative noise. A new approach of envelope extraction algorithm based on wavelet for cardiac sound signal segmentation, the new approach is based on the Morlet wavelet to extract the energy envelope of heart sound [72]. The sound data was captured by 16-bit ADC sampling at 1MHz, a normalizer shown in equation (3) was used as the secondary pre-processing stage following the Analogue to Digital Converter stage [72].

$$x_{norm}(t) = (\frac{x(t)}{\max(x(t)}))^2, where\ x\ is\ the\ t^{th}\ sample \quad (3)$$

Wavelet analysis uses time-frequency localization features to extract dilation and translation versions, the requirement of the wavelet must be oscillatory, have amplitudes that quickly decay to zero and have at least one vanishing moment to be analysed [72]. The idea of spectral envelope in proposed in [73], the proposed spectral envelope is applied to pitch shifting and envelope preservation. To realize spectral envelope, algorithms such as linear prediction (LPC) or real cepstrum are applied [73]. Discrete cepstrum and the true envelope are subject matters which were studied towards the implementation of the spectral envelope in [73], spectral envelope can be considered a smoother version amplitude spectrum. A tangentially constrained spline for envelope estimation without the undershoots problem, that can be made optimally smooth by the proposed tangential points optimization in [74]. The quadratic spline is constrained using first derivatives, and it was applied to two numerical experiments [74]. It goes through basic theories of processing envelope of sound waveforms such as analogue to digital converter, digital signal processing, windowing and spectral analysis [75]. Linear prediction coding (LPC), cepstrum and discrete cepstrum as the study subject matter for determining spectral envelope were used [75], and discrete cepstrum was improved in such a way it penalizes too that are too steep slopes of the spectral envelope by a regularization term.

2.3.1.4. Phase Detector

A novel fast phase detection digital phase locked loop is studied in literature [76], this is achieved by generating an analytic signal through the implementation of Hillbert Transform and 16-bit CORDIC algorithm is used for all complex mathematical implementation. Two platforms were used for developing and testing the digital phase detector, the model was built on MATLAB and then further tested on an FPGA [76]. The proposed All-Digital Phase Looked Loop (ADPLL) presented in literature [76] is a fully digital unit with every component digitized allowing a full implementation in an FPGA, this digital design is presented in Figure 3 below. The difference between ADPLL compared to conventional Digital PLL is that the latter has a charge pump which is analogue by design which translates phase difference into corresponding voltage [76]. F(z) is a digital filter that eliminates noise from the input signal x(n) and Ha(z) is a digital filter that does corrections on poles that introduces instability at 30º, 90º and 150º.

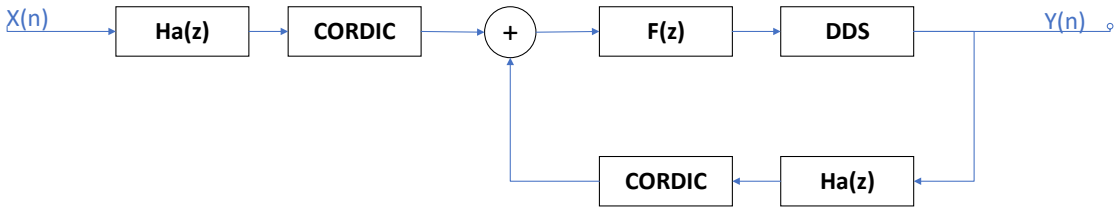


Figure 3.This portrays DSP stages for the ADPLL as presented in literature [76]

Literature [77] presents two digital phase detector algorithms that the Modified Costas Phase Detector and the CORDIC Phase Detector. The implementation of $\tan^{-1}\left(\frac{Q[n]}{I[n]}\right)$ is replaceable by its mathematical equivalent $\sin^{-1}\left(\frac{Q[n]}{\sqrt{(Q[n]^2+I[n]^2)}}\right)$, considering small input phase $\sin^{-1}\left(\frac{Q[n]}{\sqrt{(Q[n]^2+I[n]^2)}}\right)$ can be approximated by an expression $\frac{Q[n]}{\sqrt{(Q[n]^2+I[n]^2)}}$ [77]. The mathematical expression $\sqrt{(Q[n]^2+I[n]^2)}$ is the square root of the input signal power and can be removed using Automatic Gain Control (AGC), thus the Costas Phase Detector can be expressed as $\epsilon[n]=Q[n]$ [77]. The limitation of the Costas Phase Detector is that it has negative slope in the left-half of the I-Q plane, to rectify this limitation the sign of $I[n]$ is added as a correction measure making the final form of the Modified Costas Detector to be $\epsilon[n]=sgn(I[n])Q[n]$ [77]. A 180º phase-shift digital delay locked loop (DLL) study towards the design of LPDDR4 memory controllers is presented in literature [78]. Fast-locking is used as an automatic shutdown procedure to reduce power consumption [78].

A 1906 gates chip fabricated with TSMC 55nm CMOS technology Digital Phase and Frequency Detector (DPFD), the adaptation approach is applied in the fractional feedback loop to achieve highly accurate target frequency of digitally control oscillator (DCO) [79].

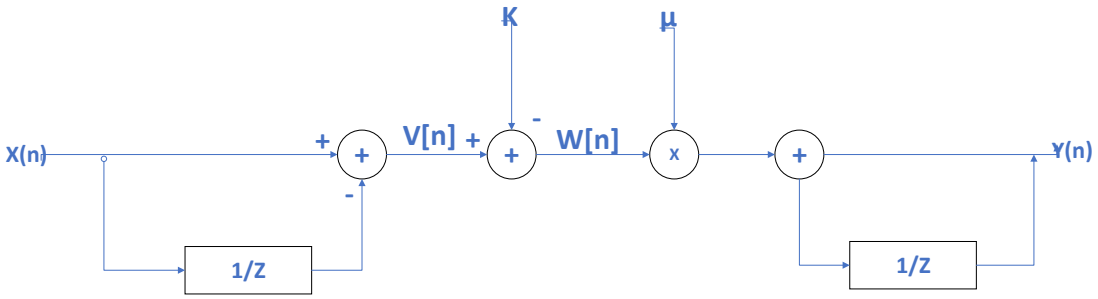


Figure 4.DPFD DSP block architecture [79]

DPFD is a two stage DSP blocks with the first stage being the phase detector $V[n]$ and the second stage being the frequency detector, the DSP equation is shown in equation 4 below.

$$Y[n] = 2\mu * \left(x[n] - {}^{x[n]}/_{Z^2} - {}^{2K}/_{Z}\right) \quad (4)$$

Table 2. Simplified Computational Requirements for DPFD [79]

Logic Slices	Quantity
Flip Flop	6
Multiplexer	6
LUT	11

2.3.2. An Overview of Analog-To-Digital Sampling Schemes and Techniques Used in This Study

The analog-to-digital converter is the heart of digitization and it is the main requirement in 4IR applications. Odd/even order sampling was studied, using step-size signal-noise-ratio (SNR), dynamic range and sampling frequency as performance parameters. Even order sampling has been applied to multiple signal sampling while achieving SNR of 64 dBFS [37]. With the rise for the need of digitization over the years other ordered sampling techniques for multiple signal sampling have been introduced [21]. In an attempt to maintain received signal power, complexities in dynamic range resolution in relation to quantization and phase errors were studied [38]. The methodology to address complexities in dynamic range resolution was

developed; it used experimental measurements of MSE on different quantization resolutions to analyze bit stream performance. The results produced indicated that the 1-bit quantization resolution has the worst phase MSE performance. To optimize digital resolution of analog-to-digital converter a quantization encoder architecture is designed [39]. Emerging attempts in addressing optimization of dynamic range resolution, has led to development of the modulo analog-to-digital converter [48] . Traditional delta-sigma sampling technique is modified in an attempt to reduce signal crosstalk in multi-channel digital converters [49]. Holistically the experiments in this dissertation were conducted using simulation tools, laboratory equipment and implementation as detailed in the points below:

Materials used in this study include:

- Laboratory equipment, Excel, simulation tool and FPGA implementation.
- Laboratory equipment such as an oscilloscope, spectrum analyzer, Radar signal synthesizer and Excel were used as preliminary efforts to develop research assumptions to a feasible study. A four channel Tektronix oscilloscope with maximum resolution of 10 million sampling points, and maximum operating frequency of 1 GHz was used to capture a sample for preliminary processing with Excel. The Tektronix spectrum analyzer with maximum operating frequency of 26 GHz was used to evaluate the quadrature signal in the time and frequency domain. The Anritsu signal generator with maximum operating frequency of 20 GHz was used to generate Radar signal. Simulation was conducted on Matlab Simulink and the delta-sigma model in [14] was used and modified to cater for the odd/even order sampling architecture. An ADL5380-EVALZ-ND I/Q demodulator was used for quadrature demodulator. Xilinx Vivado was for FPGA firmware design with MiniZed being the FPGA of choice.

Methods include:

- High level research assumptions were that odd/even order sampling will reduce total sample number and signal frequency without affecting dynamic range.
- An I/Q sine signal was designed on the Anritsu signal generator and was supplied as input to an I/Q demodulator.
- Outputs of the I/Q demodulator were connected to two channels of the oscilloscope.

- The two channels of the oscilloscope were captured and the signal was stored as comma-separated values (CSV) files for further analysis on Excel results that were plotted using Excel Figures.

Methods developed include:

- Derived mathematical model used for to process the CSV captured sine signal from the oscilloscope.
- Simulation on Matlab Simulink was used investigate the performance of step size while evaluating the effect on dynamic range.
- 1st order delta-sigma model was used in all simulation investigations and all results were plotted using Matlab Figures.
- Four sampling schemes were implemented on Xilinx Vivado, one being the proposed novel odd/even order sampling scheme and the other three were from literature.
- Performance of these schemes were captured and exported to CSV using the Integrated Logic Analyzer (ILA) from Vivado.
- CSV were further processed for SNR and results were plotted using Excel Figures.

2.4. Summary

In this chapter an attempt was made to develop a block overview of functional requirements for noise jamming architecture. Processing techniques were studied and presented in an attempt to migrate the analogue noise jamming technique. The literature study indicate an I/Q demodulator would be suitable as the initial processing block and followed by channel equalization. Both these pre-processing stages are mainly analogue while sampling, phase and frequency detector are digital processing blocks.

Chapter 3: Proposed Ordered Analog-To-Digital Sampling and Noise Sample Mixing Overview

3.2. Introduction

This chapter develops a mathematical model for odd/even order sampling from the simple form of I/Q modulator. The mathematical model caters for both quadrature signals and memory usage with strobe used to control the switch between odd/even or even/odd order sampling. Digital building blocks from previous chapter are factored further in the mathematical model. Advanced Noise jamming strategies such as Modified Shift Frequency Noise jamming, and Convolution Noise jamming were also factored into the mathematical model.

3.3. Derivation Odd/Even Ordered Analog-To-Digital Sampling and Noise Sample Mixing

The design of soft-core noise jamming architecture depends largely on the I/Q demodulator and Analogue to Digital Converter (ADC). Traditionally the design of I/Q demodulator ADC interface for Radar and Electronic Warfare (REW) applications follows five different options as depicted in [37].

3.3.1. Traditional I/Q Demodulator and Existing Even Order Sampling

Eklund and Arvidsson [37] modify the traditional I/Q demodulator and ADC interface to include an odd/even selection switch and digital filter to smooth the digital ADC output [37], this modification kicks off by declaring the complex signal and I and Q as shown in equations 1, 2 and 3 below.

$$S(t) = a_M(t) * cos\big(w_{IF} * t + \varphi_M(t)\big)$$
$$= I_{raw}(t) * \cos(w_{IF} * t) - Q_{raw}(t) * \sin(w_{IF} * t) \quad (1)$$

Table 3. Variable description for equation 1

Variable	Description
s(t)	Complex signal
I(t)	Inphase component
Q(t)	Quadrature component
W_{IF}	Intermediate Freq

$$I_{raw}(t) = a_M(t) * \cos(\varphi_M(t)) \quad (2)$$

$$Q_{raw}(t) = a_M(t) * \sin(\varphi_M(t)) \quad (3)$$

Table 4. Variable description for equations 2 and 3

Variable	Description
I_{raw}(t)	Demodulated I component
Q_{raw}(t)	Demodulated Q component
a_m(t)	Demodulated amplitude
φ_m(t)	Demodulated phase

Considering a firmware, I/ Q demodulator model with I and Q modelled by equations 2 and 3 the mixing operation will yield the results shown in Table 5. When no filtering is applied it should be noted that all the mixing coefficients results will have 1 as the maximum and -1 as the minimum.

Table 5. I/Q demodulator even 7th order digital mixing

Sample no: (n)	0	1	2	3	4	5	6	7
I-coefficients	1	0	-1	0	1	0	-1	0
Q-coefficients [1]	0	1	0	-1	0	1	0	-1

Eklund and Arvidsson [37] and Slyusar [21] applied an FIR filter to the digital firmware mixer to obtain the final form of the presented I/Q digital mixer. It should be noted that [21] implements a 4 order FIR filter instead of a 7 order FIR filter.

Table 6. *I/Q demodulator FIR filtered even 7th order digital mixing*

Sample no: (n)	0	1	2	3	4	5	6	7
I-coefficients	1	0	-11	0	15	0	-5	0
Q-coefficients [1]	0	5	0	-15	0	11	0	-1

Eklund and Arvidsson et al [37] and Slyusar et al [21] applied an FIR filter to the digital firmware mixer to obtain the final form of the presented I/Q digital mixer, it should be noted that [21] implements a 3 order FIR filter instead of a 7 order FIR filter. Slyusar [21] also introduces us to the idea of odd order sampling which is developed from even order sampling in [37], although limited theory derivation of odd order sampling he could apply it to single channel ADC I/Q sampling which uses 3rd order FIR filtering.

$$I_{even} = \cos(w_{IF} * (T_0 + \Delta) + \varphi_{IF})) * C_1 - \cos(w_{IF} * (T_2 + \Delta) + \varphi_{IF}) * C_{11} + \cos(w_{IF} * (T_4 + \Delta) + \varphi_{IF}) * C_{15} - \cos(w_{IF} * (T_6 + \Delta) + \varphi_{IF}) * C_5 \quad (4)$$

$$Q_{odd} = \sin(w_{IF} * (T_1 + \Delta) + \varphi_{IF})) * C_5 - \sin(w_{IF} * (T_3 + \Delta) + \varphi_{IF}) * C_{15} + \cos(w_{IF} * (T_5 + \Delta) + \varphi_{IF}) * C_{11} - \cos(w_{IF} * (T_7 + \Delta) + \varphi_{IF}) * C_1 \quad (5)$$

Table 7. *Variable description for equation 4 and 5*

Variable	Description
$I_{odd}(t)$	Odd sampled I component
$Q_{even}(t)$	Even sampled Q component
$T_{1,3}(t)$	Odd sampling time
$C_{1,3}$	FIR coefficients
$T_{2,4}(t)$	Even sampling time
Δ	Quantization error

3.3.2. Extension of Existing Odd Order Sampling

Before we dive down to the methodology and odd/even sampling which considers of memory usage, we need to develop the theory of odd sampling further by deriving the odd sampling expressions. When the 3rd order FIR filter in Table 4 is applied to equation 4 and 5 we get the derived equations 6 and 7 as presented below.

Table 8. *I/Q demodulator odd 4th order digital mixing*

Sample no: (n)	0	1	2	3	4
I-coefficients	0	1	0	-1	0
Q-coefficients [1]	0	0	-1	0	1

$$I_{odd} = \cos(w_{IF} * (T_1 + \Delta) + \varphi_{IF})) * C_1 - \cos(w_{IF} * (T_3 + \Delta) + \varphi_{IF}) * C_3 \quad (6)$$

$$Q_{even} = -\sin(w_{IF} * (T_2 + \Delta) + \varphi_{IF})) * C_3 + \sin(w_{IF} * (T_4 + \Delta) + \varphi_{IF}) * C_1 \quad (7)$$

The I/Q modulator of the odd order presented in [21] did not clearly derive the odd order 3rd odder characterization equation presented in the paper and equations 8 and 9 below. The derivations from 4 to 9 pave the way for the proposed odd/even algorithm that considers of memory.

$$\mathrm{I} = u_1 - 3u_3 \; where \; u_1 = \cos(w_{IF} * (T_1 + \Delta) + \varphi_{IF})) * C_1 \; is \; the \; ADC \; samples \quad (8)$$

$$\mathrm{Q} = -(3u_2 - u_4) \; where \; u_2 = \cos(w_{IF} * (T_2 + \Delta) + \varphi_{IF})) * C_3 \; is \; the \; ADC \; samples \quad (9)$$

Table 9. *4th order FIR filtering I/Q demodulator odd order digital mixing*

Sample no: (n)	0	1	2	3	4
I-coefficients	0	1	0	-3	0
Q-coefficients [1]	0	0	-3	0	1

To recap the literature sources which introduced the idea of even and odd order sampling were studied and characterization equations.

3.3.3. Derivation of Odd/Even Order Sampling

Both the even and odd order sampling algorithms do even and odd selection before the ADC changes the wave to digital making the previous algorithms implement pre-mixing. The issue with mixing instead of sample selection is that mixing makes a filter stage a requirement after

its implementation; the other issue with mixing is that mixers are usually hardware components which are not easy to digitize even in FPGA's internal mixing is done with internal PLL. It should be noted that previous approaches do not consider even and odd sample management.

The proposed methodology aims to achieve the following:

- "Elimination of the odd and even hardware mixing stages which introduce filtering complexities"
- "Introduction of flexibility that allows a single I/Q demodulator to select between even and odd"
- "Introduction of the use of fast memory at rate of 4 to 5 Megabits/s to the design of I/Q demodulators and to manage the flexibility between even and odd"

Let's consider the design of the I/Q demodulator that flexibly selects between even and odd modes at will. Further consideration is that, signal complications must also be assumed as non-existent. This assumption is valid as the I/Q hardware has improved drastically over the years. This improvement in I/Q demodulator hardware has reached a level where such devices can be bought with operational specifications such as amplitude and phase imbalance of 0.07 dB and 0.2 ˚ respectively.

Let's also consider the dual port ram modelling term $dRM_{wr_s}^{rd^s}$ where rd^s is the read memory command, wr_s is the write memory command and s is the strobe command that selects between even or odd order. Let's further consider two terms that directly tune ADC sampling, such terms are $odd\_even\_adc\_tuning$ and $even\_odd\_adc\_tuning$ as shown in Figure 5 below.

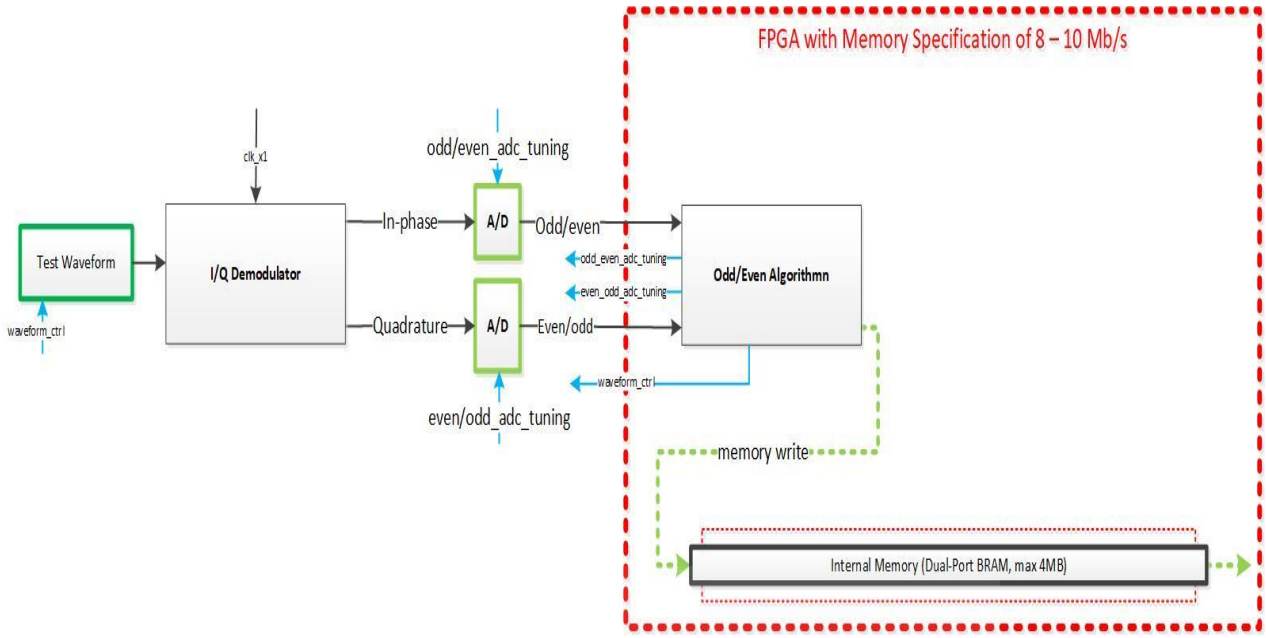


Figure 5. The proposed ordered sampling model

Table 10. *I/Q demodulator odd 7th order digital mixing*

Sample no: (n)	0	1	2	3	4	5	6	7
I-coeff	0	1	0	-1	0	1	0	-1
Q-coeff [1]	0	0	-1	0	1	0	1	0

Since the proposed algorithm excludes the signal filtering stage, the filter phase Δ and coefficients C_n terms in equation 4 to 7 fall away. When we apply the ADC tuning terms which are controlled by the memory strobe the form presented in equation 4 becomes the new equation 10 below:

$$I_{odd} = \{(odd\_even\_adc\_sampling = 1)|\{\cos(w_{IF} * T_1 + \varphi_{IF}) - \cos(w_{IF} * T_3 + \varphi_{IF}) + \cos(w_{IF} * T_5 + \varphi_{IF}) - \cos(w_{IF} * T_7 + \varphi_{IF})\}\} \quad (10)$$

$$I_{odd} = \sum_{S_i=0}^{N} \{(odd\_even\_adc\_sampling = 1)|I_{raw}(S_i)\} \quad (11)$$

It should be noted that the algorithm presented in equations 10 and 11 selected odd or even sampling by using $odd\_even\_adc\_tuning$ which is controlled by the dual port memory.

$$Q_{even} = \{(odd\_even\_adc\_sampling = 1)| \{\sin(w_{IF} * T_2 + \varphi_{IF}) - \sin(w_{IF} * T_4 + \varphi_{IF}) + \sin(w_{IF} * T_6 + \varphi_{IF})\}\} \quad (12)$$

$$Q_{even} = \sum_{S_i=0}^{N} \{(odd\_even\_adc\_sampling = 1)|Q_{raw}(S_i)\} \quad (13)$$

Equations 10, 11, 12 and 13 built up the derivation to the final form of the proposed odd order sampling with memory consideration. The full form of the proposed odd order sampling is presented in equation 14. The full form of this algorithm selects odd sampling by using s which controls the ADC sampling and dual port memory write location.

$$dRM_{wr_s}^{rd^s} = \sum_{s_i=0}^{N} \left\{ \{(wr_s = 1 \; And \; s = 00)?\}| \left\{ \left\{ (s_i \; Mod \; 2 = 1)?| \sum_{i_s=0}^{255} I_{odd}(S_i), \right\} | \left\{ \{(s_i \; Mod \; 2 = 0)?| \sum_{q_s=265}^{520} Q_{even}(S_i), \right\} \right\} \right\} \quad (14)$$

Table 11. *Odd order sampling proposed memory mapping*

Input Addr	Dual-Port Mem	Output Addr
0x00	I (0)	0x00
…	…	…
0xFF	I (i)	0xFF
Reserved to guard data leak + 2-bit strobe		
0x109	Q (0)	0x109
…	…	…
0x208	Q (i)	0x208

Assuming sampling window of N samples as shown in equation 13, to store such a window a 2-bit strobe shown in Table 12 is used to tune the ADC in such a way that either the combination I_{odd}, Q_{even} or I_{even}, Q_{odd} as shown in Tables 11 and 14. The memory is mapped in such a way that 8 bits addressing is achieved, furthermore the memory data leak has been

accounted for by introducing guard gaps. 8 bits addressing, storage of I_{odd} and Q_{even} for odd order sampling and guard gaps are presented in Table 11.

Table 12. *Bit arrangement at memory located 0x104*

Bit No	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Bit Function	**Strobe**		Reserved	Reserved	Reserved	Reserved	**Flag**	

$$I_{even} = \{(odd\_even\_adc\_sampling = 0)\ |\{\cos(w_{IF} * T_0 + \varphi_{IF}) - \cos(w_{IF} * T_2 + \varphi_{IF}) + \cos(w_{IF} * T_4 + \varphi_{IF}) - \cos(w_{IF} * T_6 + \varphi_{IF})\}\} \quad (15)$$

$$I_{even} = \sum_{S_i=0}^{N} \{(odd\_even\_adc\_sampling = 0)|I_{raw}(S_i)\} \quad (16)$$

$$Q_{odd} = \{(even\_odd\_adc\_sampling = 0)\ |\{\sin(w_{IF} * T_1 + \varphi_{IF}) - \sin(w_{IF} * T_3 + \varphi_{IF}) + \sin(w_{IF} * T_5 + \varphi_{IF}) - \sin(w_{IF} * T_7 + \varphi_{IF})\}\} \quad (17)$$

$$Q_{odd} = \sum_{S_i=0}^{N} \{(odd\_even\_adc\_sampling = 0)|Q_{raw}(S_i)\} \quad (18)$$

$$dRM_{wr_s}^{rd^s} = \sum_{s_i=0}^{N} \left\{ \{(wr_s = 1\ And\ s = 01)?\}| \left\{ \left\{ (s_i\ Mod\ 2 = 0)?| \sum_{i_s=274}^{871} I_{even}(S_i), \right\} | \left\{ (s_i\ Mod\ 2 = 1)?| \sum_{q_s=3}^{258} Q_{odd}(S_i), \right\} \right\} \right\} \quad (19)$$

Table 13. *Variable description for equations 14 and 19*

Variable	**Description**
dRM	Dual-port memory
wr_s	Memory write
rd^s	Memory read
S_i	Sample number
s	Memory strobe

3.3.4. Derivation of Odd/Even Order Sampling Memory Management and Envelope Detection

Equations 15, 16, 17 and 18 built up the derivation to the final form of the proposed even order sampling with memory consideration. The full form of the proposed even order sampling is presented in equation 19. The full form of this algorithm selects odd sampling by using *s* which controls the ADC sampling and dual port memory write location.

Table 14. E*ven order sampling proposed memory mapping.*

Input Addr	Dual-Port Mem	Output Addr
0x03	Q (0)	0x03
…	…	…
0x102	Q (i)	0x102
Reserved to guard data leak + 2-bit strobe		
0x112	I (0)	0x112
…	…	…
0x367	I (i)	0x367

8 bits addressing, storage of I_{even} and Q_{odd} for even order sampling and guard gaps are presented in in Table 14. Dual-port RAM memory has unique properties that allow the stored data to be read in consecutive clock cycles. This means theoretically an extremely fast processing throughput can be achieved. The proposed envelope detection algorithm in [70] divides the signal $S(t)$ to into k bunches of N samples.

$$I_{odd} = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} \{dRM_{wr_s=0}^{rd^s=1}(addr)\} \tag{20}$$

$$Q_{even} = \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} \{dRM_{wr_s=0}^{rd^s=1}(addr)\} \tag{21}$$

$$S_{256} = \sum_{i=1}^{256} S(i) = \sum_{i=1}^{128} I_{odd}(i) + \sum_{i=1}^{128} Q_{even}(i) \tag{22}$$

$$S_{256} = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} \{dRM_{wr_s=0}^{rd^s=1}(addr)\} + \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} \{dRM_{wr_s=0}^{rd^s=1}(addr)\} \tag{23}$$

Equations 20, 21 and 22 built up the derivation to the final form of the proposed memory read algorithm. The full form of the memory read algorithm is presented in equation 23. The first

step in [70] takes the absolute value of $S(t)$, since we have modified $S(t)$ to 256 samples then we have taken the absolute value of S_{256} as shown in equations 24 and 25.

$$|S_{256}| = \sum_{i=1}^{256} |S(i)| = \sum_{i=1}^{128} |I_{odd}(i)| + \sum_{i=1}^{128} |Q_{even}(i)| \tag{24}$$

$$|S_{256}| = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} |dRM_{wr_s=0}^{rd^s=1}(addr)| + \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} |dRM_{wr_s=0}^{rd^s=1}(addr)| \tag{25}$$

The second step in [70] divides the signal $S(t)$ into k bunches of N samples, since we have modified $S(t)$ to 256 samples then we have divided S_{256} in k_m bunches of 256 samples. k_m is a memory address with equivalent size to k and the full k_m memory sorting algorithm as presented in equations 26 and 27.

$$|S_{256}| = \sum_{i=1}^{k} |S(i)| = \sum_{i=1}^{k/2} |I_{odd}(i)| + \sum_{i=1}^{k/2} |Q_{even}(i)| \tag{26}$$

$$|S_{256}| = \sum_{clk=2}^{(k+2)/2} \sum_{addr=0x00}^{k_m} |dRM_{wr_s=0}^{rd^s=1}(addr)| + \sum_{clk=3}^{(k+3)/2} \sum_{addr=0x633}^{k_m} |dRM_{wr_s=0}^{rd^s=1}(addr)| \tag{27}$$

The third and the last step in [70] checks for maximum peaks and modifies each element in the k_m set to the maximum peak as presented in equations 28 and 29.

$$I_{env} = \sum_{clk=2}^{(k+2)/2} \sum_{addr=0x00}^{k_m} \Big\{ |\big(dRM_{wr_s=0}^{rd^s=1}(addr)| > |dRM_{wr_s=0}^{rd^s=1}(addr-1)|\big) ? |\{|dRM_{wr_s=1}^{rd^s=0}(addr-1)| = |dRM_{wr_s=0}^{rd^s=1}(addr)|\}\Big\} \tag{28}$$

$$Q_{env} = \sum_{clk=3}^{(k+3)/2} \sum_{addr=0x633}^{k_m} \Big\{ |\big(dRM_{wr_s=0}^{rd^s=1}(addr)| > |dRM_{wr_s=0}^{rd^s=1}(addr-1)|\big) ? |\{|dRM_{wr_s=1}^{rd^s=0}(addr-1)| = |dRM_{wr_s=0}^{rd^s=1}(addr)|\}\Big\} \tag{29}$$

Table 15. *Variable description for equations 28 and 29*

Variable	Description
addr	Dual-port memory address
k	Peak detection bunch size
K_m	Peak detection mth element
I_{env}	I component of envelope
Q_{env}	Q component of envelope

The odd/even sampling on I/Q signals. The signal $S(t)$ when applied to an I/Q demodulator signals S_{I_odd} and S_{Q_even} are produced. The introduction of $S_{I\_odd}(t)$ and $S_{Q\_even}(t)$ allow for k bunches to be taken from N samples and odd/even sampling reduces a N I/Q to $N/2$ I/Q signal samples. The memory reading procedure allows for both odd and even sample data to be read, from clock cycle 2 to 257 odd sample data is read as presented in equation 18 and 19, at 269 to 525 even sample data is read as presented in equations 20 to 21. The memory reading procedure allows for both odd and even sample data to be read, from clock cycle 2 to 257 odd sample data is read as present.

3.3.5. Application of Modified Costas to Memory Samples to Achieve Angle Detection

Literature [77] in presents the idea of modified the Costas phase detector given by equation 30 with phase detector gain given by equation 31, the modified Costas phase detector can be improved to phase detect odd/even sampled I/Q signals stored in memory as shown in equation 32 and 33. The power term $\sqrt{P_{in}}$ in equation 34 can be replaced by the envelope I_{env} or Q_{env} to get equation 35.

$$\epsilon[n] \quad = sgn(I[n]) * Q[n] \tag{30}$$

$$k_D \quad = \frac{\sqrt{P_{in}}\sin(\epsilon[n])}{\epsilon[n]} \tag{31}$$

$$\epsilon_{128} = \sum_{i=1}^{128} sgn(I_{odd}(i)) * \sum_{i=1}^{128} Q_{even}(i) \tag{32}$$

$$\epsilon_{128} = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} sgn\left(dRM_{wr_s=0}^{rd^s=1}(addr)\right) * \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} dRM_{wr_s=0}^{rd^s=1}(addr) \tag{33}$$

Equations 30, 31 and 32 built up the derivation to the final form of the proposed modified Costas phase detector algorithm with memory considerations. The full form of the modified phase detector algorithm with memory read is presented in equation 33.

$$K_{D_{128}} = \sum_{i=1}^{128} \frac{\sqrt{P_{in}(i)}\,\sin(\epsilon_{128}(i))}{\epsilon_{128}} \tag{34}$$

$$K_{D_{128}} = \sum_{i=1}^{128} \frac{I_{env}(\mathrm{i}) * \sin(\epsilon_{128}(i))}{\epsilon_{128}} \tag{35}$$

Subsection 3.3.3 introduced the idea of odd/even sampling on I/Q signals. The signal $S(t)$ when applied to an I/Q demodulator signals S_{I_odd} and S_{Q_even} are produced. The introduction of $S_{I\_odd}(t)$ and $S_{Q\_even}(t)$ allow for k bunches to be taken from N samples and odd/even sampling reduces a N I/Q to $N/2$ I/Q signal samples. The memory reading procedure allows for both odd and even sample data to be read, from clock cycle 2 to 257 odd sample data is read as presented in equation 18 and 19, at 269 to 515 even sample data is read as presented in equation 20 to 21. The memory reading procedure allows for both odd and even sample data to be read, from clock cycle 2 to 257 odd sample data is read as present.

3.3.6. Application of Discrete Fourier Transforms(DFT) to Achieve Frequency Detection

The Discrete Fourier Transforms (DFT) a sequence of N complex samples in time domain $\boldsymbol{x_n} = x_0, x_1, \ldots\ldots\ldots, x_{N-1}$ into another sequence of complex samples in frequency domain $\boldsymbol{X_k} = X_0, X_1, \ldots\ldots\ldots, X_{N-1}$ as defined in equation 36.

$$X_k = \sum_{n=0}^{N-1} x_n * e^{-\frac{i2\pi}{N}kn} = \sum_{n=0}^{N-1} \left[\cos\left(\frac{2\pi kn}{N}\right) - i * \sin(\frac{2\pi kn}{N})\right] \tag{36}$$

Our goal is to conduct the spectral analysis of S_{256} dual-port memory stored complex signal, then x_n becomes the signal of interest which is S_{256}. Equation 37 describe the updated form of equation 36:

$$X_k = \sum_{n=1}^{256} S_{256} * e^{-\frac{i2\pi}{256}kn} = \sum_{n=1}^{256} \left[\cos\left(\frac{2\pi kn}{256}\right) - i * \sin(\frac{2\pi kn}{256})\right] \tag{37}$$

$$X_k = \sum_{n=1}^{256} S_{256} * e^{-\frac{i2\pi}{256}kn} = \sum_{n=1}^{128} I_{odd} * e^{-\frac{i2\pi}{128}kn} + \sum_{n=1}^{128} Q_{even} * e^{-\frac{i2\pi}{128}kn} \quad (38)$$

Equations 36, 37 and 38 built up the derivation to the final form of the proposed modified Discrete Fourier Transform (DFT) algorithm with memory considerations. The full form of the modified DFT algorithm with memory read is presented in equation 39.

$$X_k = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} \sum_{n=1}^{128} \{dRM_{wr_s=0}^{rd^s=1}(addr) * e^{-\frac{i2\pi}{128}kn}\} + \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} \sum_{n=1}^{128} \{dRM_{wr_s=0}^{rd^s=1}(addr) * e^{-\frac{i2\pi}{128}kn}\} \quad (39)$$

3.3.7. Derivation of Noise Sampling Mixing through Modified Time Delay

A modified time delay presented by Xun, Cui and Hao [7] is concerned with time delay echo with inter-pulse jitter. Their research shows that time-delay echo jamming can only generate false information if it retains the signal inter-intra phase characteristics. The transmitted by a Linear Frequency Modulation (LFM) Radar can be shown by equation 1, where $a_M(t)$ is signal amplitude which is best described by the envelope detector equation 28 and 29. w_{IF} is the frequency modulation frequency and $\varphi_M(t)$ is the inter-intra phase. The time-delay echo jammed signal is given by equation 40:

$$S(t) = a_M(t - \tau_d) * cos\big(w_{IF} * (t - \tau_d) + \varphi_M(t)\big) \quad (40)$$

$$\tau_d(t) = \tau_{d0} + \tau_\xi(t) \quad (41)$$

The time-delay factor τ_d varies from pulse to pulse and the expression is given by equation 41, where τ_{d0} is a constant time delay which should not be less than the minimum echo time delay of the jammer. $\tau_\xi(t)$ is a random variable within a fast time range signed as $[\tau_1, \tau_2]$, if T is the transmitting pulse width then literature [7] has a constraint of $\tau_2 - \tau_1 < T$.

$$\tau_{addr}(t) = \tau_{addr0} + \tau_\xi(t) \quad (42)$$

$$S_{256}(\tau_d) = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} \{dRM_{wr_s=0}^{rd^s=1}(addr + \tau_{addr})\} + \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} \{dRM_{wr_s=0}^{rd^s=1}(addr + \tau_{addr})\} \quad (43)$$

Sample time information is locked on memory address, to make τ_d for memory stored discrete time samples then τ_d must be modified to memory delay τ_{addr}. $\tau_\xi(t)$ is a modified random memory access address within range signed as $[\tau_{addr1}, \tau_{addr2}]$; if T_{addr} is the stored pulse width and the modified operating constraint of $\tau_{addr2} - \tau_{addr1} < T_{addr}$. The final form for block-ram memory access model for noise generator obtained by modified time delay is presented in equation 43.

3.3.8. Derivation of Noise Sampling Mixing through Modified Shift Frequency

It is well known that shift frequency jamming derived from signal frequency characteristic is effective against Linear Frequency Modulation (LFM) Radar [8]. Few research outputs discuss the mathematical model that interface shift frequency with cyclic dual-port memory in pulse-doppler Radar, for this interface to be possible it important to modify the modified shift frequency equations in such a way 256 sample memory stored LFM Radar waveform can be used. The signal transmitted by a Linear Frequency Modulation (LFM) Radar can be shown by equation 1, where $a_M(t) = 1$ as signal amplitude, w_{IF} as the frequency modulation frequency and $\varphi_M(t)$ is the inter-intra phase. The shift frequency jammed signal is given by equation 44:

$$s(t) = cos\big((w_{IF} + w_d) * t + \varphi_M(t)\big) \quad (44)$$

To effectively modify the shift frequency mathematical model, it is necessary to transform equation 44 to frequency domain using the DFT introduced in equation 36 and when applied to equation 44 we get equation 45 below:

$$S_k = \sum_{n=0}^{N-1} s_n * e^{-\frac{i2\pi}{N}kn} \quad (45)$$

$$w_d = \xi * B_r \quad (46)$$

Sample frequency information is locked as DFT of memory address lock time information, to make w_d for memory stored discrete frequency samples then w_d must be modified to memory shift w_{addr}. ξ is a modified random memory access address within range signed as $[a, b]$, if

B_r is the LFM Radar bandwidth. The equation 47 for ω_{addr} has the sampled LFM Radar bandwidth M_r. The final form for block-ram memory access model for noise generator obtained by modified shift frequency is presented in equation 48.

$$\omega_{addr} = \xi * M_r \tag{47}$$

$$S_k = \sum_{clk=2}^{130} \sum_{addr=0x00}^{0x78} \sum_{n=1}^{128} \{dRM_{wr_s=0}^{rd^s=1}(addr + \omega_{addr}) * e^{-\frac{i2\pi}{128}kn}\} + \sum_{clk=3}^{131} \sum_{addr=0x633}^{0x75B} \sum_{n=1}^{128} \{dRM_{wr_s=0}^{rd^s=1}(addr + \omega_{addr}) * e^{-\frac{i2\pi}{128}kn}\} \tag{48}$$

3.3.9. Derivation of Noise Sampling Mixing through Convolution

Lv [9] claims noise convolution jamming or smart noise jamming technique, this technique receive part of the LFM Radar process gains and this makes superior to existing noise blanketing techniques. The main advantage of noise convolution jamming is that it reduces jamming energy and improve jamming power utility greatly [9]. In operation noise convolution selects the initial and terminal time to control the position and width of jamming region to suppress distributed target [9]. J. Jiang, Y. Wu, H. Wang et al [80] reintroduces the ideas of RF noise jamming, noise AM jamming, noise FM jamming and noise convolution for jamming of ground targets. The results show that noise convolution jamming out performs other non-coherent jamming techniques [80] .Barrage noise jamming is by far the most important type of jamming for airborne phase array Radar [81]. [82]Few research outputs discuss the mathematical model that interface noise convolution with cyclic dual-port memory in pulse-doppler Radar. In operation scenario the jammer receives the radar's emission signal after time delay $\tau = {R_j(s)}/{c}$ [9]. $R_j(s)$ is the slant range between jammer and radar, it's a function of azimuth slow time s; while t is the range fast time and $c = 3 * 10^8\ m/s$ is the velocity of propagation of electromagnetic waves in vacuum [9]. When jammer receives radar's emission signal it applies convolution with noise, afterwards amplifies and retransmits the jammed radar signal [9]

$$S_{conv} = f(t) * S(t) \tag{49}$$

Where $S(t)$ in equation 49 is the radar signal that the jammer received, the radar is eventually odd/even sampled and get stored to dual-port cyclic memory as S_{256}. Equation 49 becomes equation 50 after ADC and memory stage.

$$S_{conv_{256}} = f(t) * S_{256} \tag{50}$$

To conduct convolution on 256 sample radar wave it is necessary to understand the concept of single pulse convolution given by equation 51 in time domain and equation 52 in discrete time domain. Equation 51 is an impulse with amplitude A_1 and time delay t_1 in discrete time the discrete time delay become i_1. When jammer discovers radar signal it convolves the impulse with radar signal to get equation 53:

$$p(t) = A_1\delta(t - t_1) \tag{51}$$

$$p = \sum_{i=1}^{N} A_1\delta(i - i_1) \tag{52}$$

$$S_{impulse} = p(t) * S_{conv_{256}} = \sum_{i=1}^{246} A_1 S_{256}(i - i_1) \tag{53}$$

Let suppose that the noise used for convolution is defined in equation 54,

$$f(t) = U_n(t)e^{j\phi(t)} \tag{54}$$

The noise envelope is defined by U_n and it obeys various noise model distributions such as Rayleigh, Gaussian and uniform distributions, $\phi(t)$ is the phase function. The noise signal $f(t)$ has no deterministic expression and it is impossible to conduct mathematical derivation, but we can regard it as composed of finite number of pulses with different amplitudes and delays. We need to limit the pulse delay time in equation 51 and 52, this is to set limit to pulse position according equation 55. [83, 84, 85, 86, 87] use noise models such as Rayleigh, Gaussian and uniform distribution to model the behaviour of digital systems in excessive noise environment. [88, 89] simulates and statistical analysis of Radio communication and Radar systems with radio noise on Radio communication systems, on Radar system convolution modulation.

$$0 \leq t_1 \leq {\left(2\left[R_{max} - R_j(s)\right]\right)}/{c} \tag{55}$$

Where R_j is the space between the jammer and Radar while R_{max} is the maximum range of the Radar. Solving for equation 53 and 54 we get equation 56 below with 56 being the final form for noise convolution algorithm:

$$S_{impulse} = \sum_{i=1}^{246} A_1 S_{256}(i - i_1) \; = \sum_{i=1}^{256} U_{256}(i) e^{j\Phi(i)} * S_{256}(i - i_1) \tag{56}$$

3.4. Simulation and Experimental Requirements for Ordered Analog-to-Digital Sampling

EW components which are I/Q Demodulator, Channel Equalizer, Envelope Detector, Phase Detector and Frequency Detector as envisioned in the architecture breakdown. Derivation of the model on previous section covered each of these components, these are common EW building blocks and Soft-Core Architecture. Through derivation of the model a numerical issue of memory data rate constraint was deduced. Literature and numerical investigation on I/Q demodulators led to the development of odd/even order sampling which only stores into memory odd samples from the I channel and even samples from the Q channel. The proposed architecture has a challenging memory requirement of 8 to 10 Mb/s as depicted in Table 18. With the development of odd/even order sampling an FPGA with 4 to 5 Mb/s is an organic choice for performance specification requirement. Although odd/even order sampling has been applied for digital communication applications, no satisfactory methods combining odd/even order sampling with soft-core noise jamming architecture have been developed so far.

Envelope, phase and frequency detectors plus their literature fall under the sub-field of Digital Signal Processing (DSP). DSP mainly uses number crunching to implement equations and algorithms. Different equations and empirical algorithms to calculate envelope and phase of a complex signal were explored in the previous section, different strategies were explored and all of them as an algorithm. Discrete Time Fourier Transform (DTFT) has become the main DSP algorithm for converting discrete time sample to frequency domain. The specifications for most DSP equations and algorithms cannot be easily determined as the selected FPGA becomes a constraint to the processing requirements of any DSP algorithms, and the selection of FPGA for this study largely depends on the data rate for memory shown in Table 18.

Table 16. *Component category form or purpose matrix for the proposed soft-core architecture with each component allocated to a form or purpose.*

Soft-Core Functionality	Firmware	Hardware	Measuring
Signal Synthesis			X
Demodulation		X	
Digitization	X		
Parameter Storage	X		
Envelope Extraction	X		
Phase Extraction	X		
Freq Extraction	X		
Noise Synthesis	X		
Test & Evaluation	X		X

Table 17. *Firmware components category matrix allocates the firmware components to either Signal Conditioning, Digital Signal Processing or Radar & Electronic Warfare Signal Processing*

Firmware Component	Implementation	Detailed Design Functionality
Parameter Storage	Signal Conditioning	Odd samples of I channel are selected and store. Even samples of Q channel are selected.
Envelope(Env) Extraction	Digital Signal Processing	Recovers the pulse radar envelope for processing. This processing step allows for the radar pulse shape to be reproduced, the beginning and end of the

		pulse to be marked. This processing step also allows for transmitted radar power to be measured, reproduced and modified.
Phase Extraction	Digital Signal Processing	Recovers the modulation phase shift from the I/Q demodulator. This processing step allows for the radar complex signal properties to be measured, reproduced and modified.
Freq Extraction	Digital Signal Processing	Recovers the pulse and modulation frequencies from I/Q demodulator.
Noise Synthesis	Radar & EW Signal Processing	Generates noise and limits the noise bandwidth to a reference bandwidth of 1.5kHz with pulse magnitude from RAM Access component

Different equations and empirical algorithms to calculate noise generation, frequency detection and noise FM to produce complex noise jamming signal were explored in section A, different strategies were explored and all of them are either an equation or an algorithm. Three different strategies for noise generation which are modified time delay, modified shift frequency and convolution were discovered during noise jamming background review. The specifications for RSP equations and algorithms cannot be easily determined as the selected FPGA becomes a constraint to the processing requirements of any RSP algorithms, and the selection of FPGA for this study largely depends on the data rate for memory shown in Table 16, 17 and 18.

Table 18. *Hardware components specification requirements*

Hardware Component	Implementation	Specification Requirements
Demodulation	Signal Conditioning	Freq Range 400 MHz – 6 GHz, Amplitude Imbalance 0.07 dB, Phase Imbalance 0.2˚, Demodulation bandwidth 390 MHz
Digitization	Digital Signal Processing	Freq Range 1-5 GHz with -3 dB at 9 GHz, Resolution 12-Bit
Env, Phase, Freq Extraction	Radar & EW Signal Processing	Memory Data Rate 8-10 Mb/s(min), 1-5 GHz Analog-to-Digital Converter,
Test & Evaluation	Radar & EW Signal Processing	Integrated Logic Analyzer (ILA) Data Rate 8-10 Mb/s(min), 4Mb window(max)

3.5. Summary

A mathematical model was developed using digital building blocks from previous chapter and quadrature modulator equations. Advanced Noise jamming strategies such as Modified Shift Frequency Noise jamming, and Convolution Noise jamming were also factored into the mathematical model. Hardware and software requirements were developed for simulation and implementation purposes.

Chapter 4: Case Study 1: Experimental Verification of Sample Frequency

4.1. Introduction

This Chapter applies the derived mathematical model components derived in the previous section on potential hardware. Equipment such as signal generator was used to generate test signals. Oscilloscope was used to capture and digitize the test signals, spectral analyser was used to design and adjust signal parameters such as modulation angle, frequency and phase. The signal was analysed by applying the equations in Chapter 4 on Microsoft Excel to determine preliminary results.

4.2. Overview of Experimental Setup

This experimental setup is centred around generation of I/Q demodulator signals as the system under investigation. The experimental equipment available to conduct this experiment are presented in Table 19.

Table 19. Measurement equipment available for experimental setup

Experimental Equipment	Equipment Range
Anritsu Signal Generator	1 Hz to 20 GHz
Tektronix Oscilloscope	1 Hz to 1 GHz
Tektronix Spectral Analyzer	1 Hz to 26 GHz

Flexible waveguides were selected for interconnection between the signal source, system under investigation and measurement equipment. The procured I/Q demodulator with dual p and n channels was used throughout measurement efforts. It is important to emphasize that the procured I/Q demodulator was supplied without n channel connected on the printed circuit board (PCB). The I/Q demodulator connection schedule is given in Table 20.

Table 20. ADL5380-EVALZ-ND I/Q Demodulator Electrical Connection Schedule

Connector Schedule	Electrical Schedule
RF_p	**SMA to Waveguide from Signal Gen**
RF_n	No SMA connector PCB terminated
I_p	**SMA to Waveguide to Oscilloscope**
I_n	No SMA connector PCB terminated
Q_p	**SMA to Waveguide to Oscilloscope**
Q_n	No SMA connector PCB terminated
V_cc	**Croc Clips to Power Supply**
Ground	**Croc Clips to Power Supply**

A proposed experimental setup to investigate research assumption for the new odd/even order sampling of I/Q signals was conducted in the laboratory environment. Experiment setup to acquire I/Q signals consists of Tektronix Oscilloscope with integrated spectral analyser and 12-bit analog-to-digital converter, 5 GHz local oscillator (LO) driven by voltage-controlled oscillator (VCO) which can be configured to generate 1 Hz to 5 GHz and Anritsu Signal Generator. Anritsu signal generator was used as the signal source with 5 Ω characteristic load and maximum signal power design to 0.0 dBm. The source was configured to produce a phase modulated complex signal with I and Q separated by a constant phase shift. To verify the design of a complex, signal the Tektronix spectral analyser was used in complex function. Other analysis such as spectrogram to confirm percentage pulse overlap between I and Q was also conducted and overlap of 19% was verified as shown in Table 21.

Table 21. Spectrogram measurements results

Measurements		Settings	
Δ Overlap	**Freq**	**Time/Div**	**Span**
19%	1.004649 GHz		
		450us	24.40MHz

Table 22. High-end wave front frequency spectrum measurements results.

Measurements		Settings	
Amplitude	**Freq**	**Time/Div**	**Span**
-72.56dBm	1.004649 GHz		
		10 dB	24.40MHz

From the spectrogram window it was possible to place the marker at the signal wave front and to able to measure the high-end signal cut off in frequency domain as shown in measurement results in Table 22.

4.3. Experiment One – Collection of Signal Data

The next step in the experimental setup was to pass the confirmed complex signal to the I/Q demodulator hardware to recover complex I and Q components. I/Q demodulator operates at 5 volts and power supply was set to 5 volts and it was confirmed to draw 200 mA of current. Preliminary results on the performance of the procured I/Q demodulator are given in Figure 6.

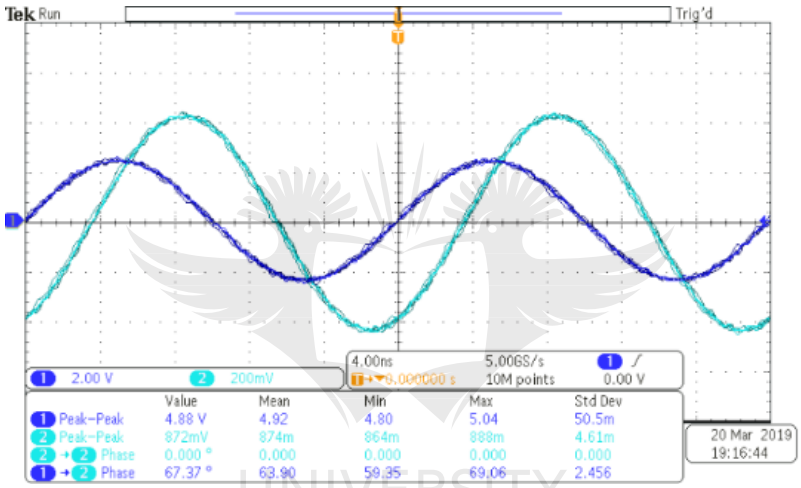


Figure 6. This presents oscilloscope output performance of I/Q demodulator with I component connected to Channel 1 and Q connected to Channel 2.

A complex signal from the signal generator was passed through I/Q demodulator, and outputs of the I/Q demodulator were passed to the oscilloscope for analysis. Analysis results reveal that Q component lags I component by 63.90˚ as shown in Table 23 and Figure 6.

Table 23. I/Q demodulator performance measurements results

	Measurements		Settings	
	Ch1	**Ch2**	**Ch1**	**Ch2**
Amplitude(P-P)	4.92 V	874 mV	2V/div	200mV/div
Phase	63.90˚	0.00˚	4ns/div	4ns/div

4.4. Experiment Two – Analysis of Signal Data

The initial step in approaching the analysis of the captured signal wave data is to verify that the exported CSV signal data is valid. The signal wave data acquired in Figure 6 was reproduced in Excel; the phase and amplitude of Excel reproduced data aligns with the one presented in Figure 8, Excel reproduced signal data is presented in Figure 7. It is important to note that the reproduced signal is limited to 256 samples, this is to ease processing on Excel.

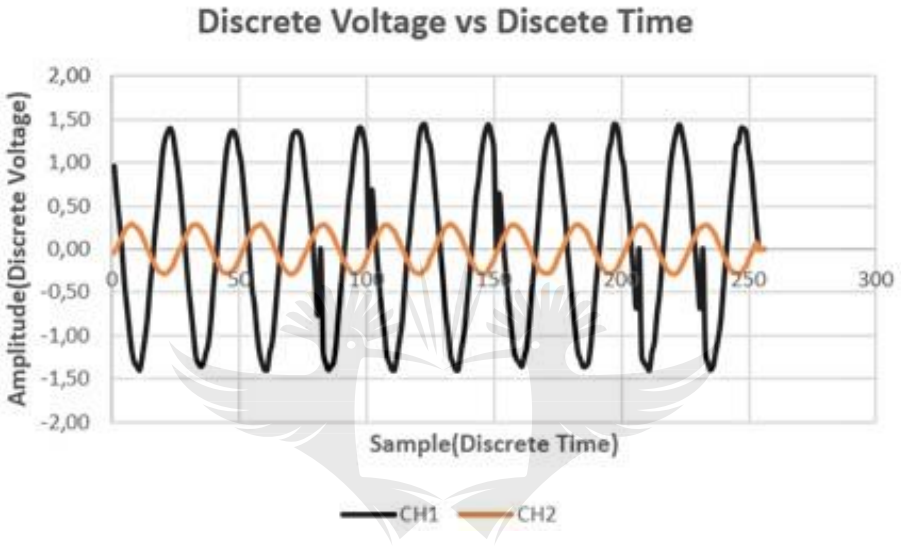


Figure 7. *This presents Excel reproduced acquired I/Q demodulator samples*

The new odd/even order sampling mathematical model in equations 12 and 13 is applied afterward Fourier Transform Analysis was applied to obtain the sample frequency behaviour shown in Figure 8. Results in Figure 8 validates the research assumption that the new odd/even order sampling is half the sampling frequency.

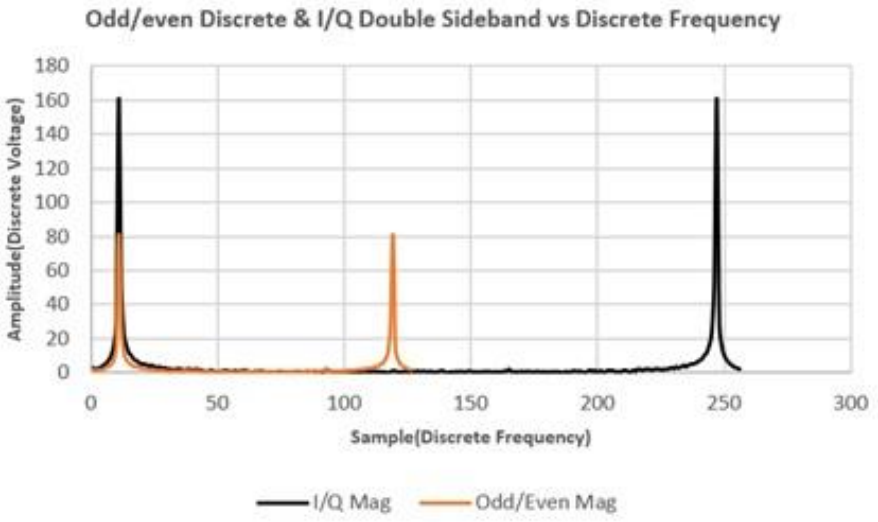


Figure 8. This presents new odd/even order sample frequency behaviour

A separate investigation onto effects of the new odd/even sampling onto phase behaviour have indicated that it suffers from phase error with I sampling component affected more than the Q sampling component as shown in Figure 9. It is not clear what causes this behaviour but literature [1] have recommended the use of FIR filter to correct this issue.

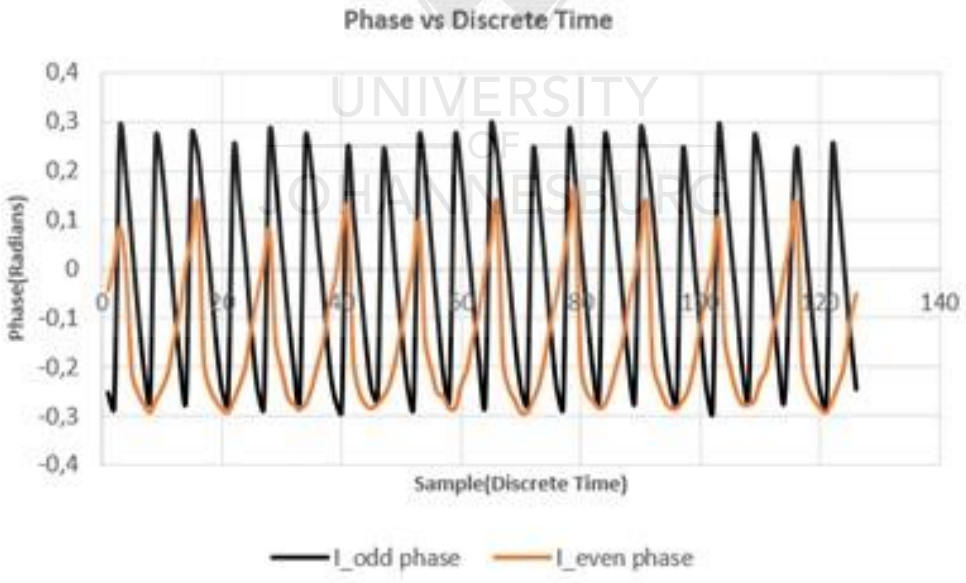


Figure 9. *This presents phase error behavior of the new odd/even order sampling*

Peak detect algorithm derived in equations 28 and 29 was applied to waveform data obtained in the previous section. According to the proposed new algorithm for envelope detection with memory considerations the sample radar waveform is normalized to only positive values and

then segmented into K segments. All other elements in the K segments are replaced by the maximum peak to give the envelope in Figure 10. The product can be made smoother by introducing a low pass filters but this remains out of scope for this work.

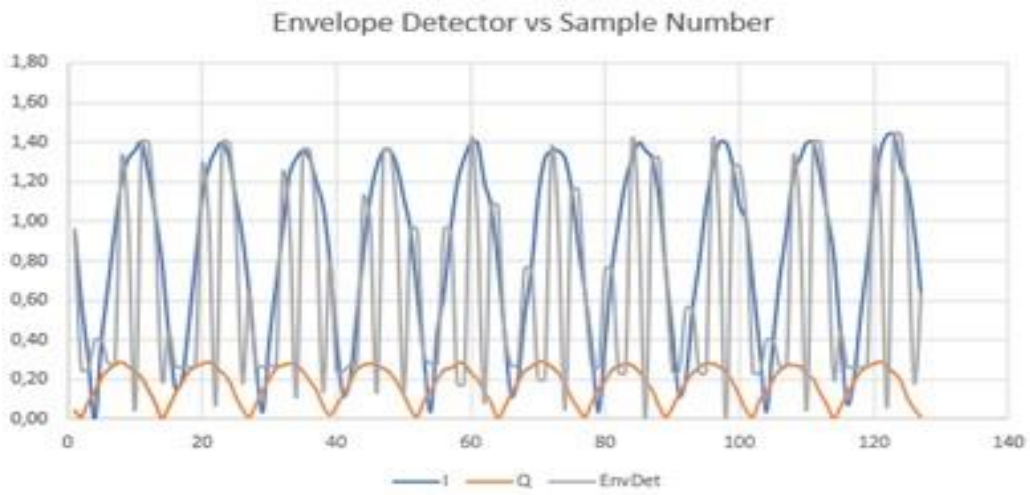


Figure 10. *Envelope generated by the new peak detection envelope detector with memory consideration, in this test scenario K=3*

Modified Costas phase detector derived in equations 32 and 33 was also applied to the same waveform data in Figure 6. The proposed new modified Costas phase detector with memory considerations, in sample manner calculates reliable phase error with limited processing power as shown in Figure 11.

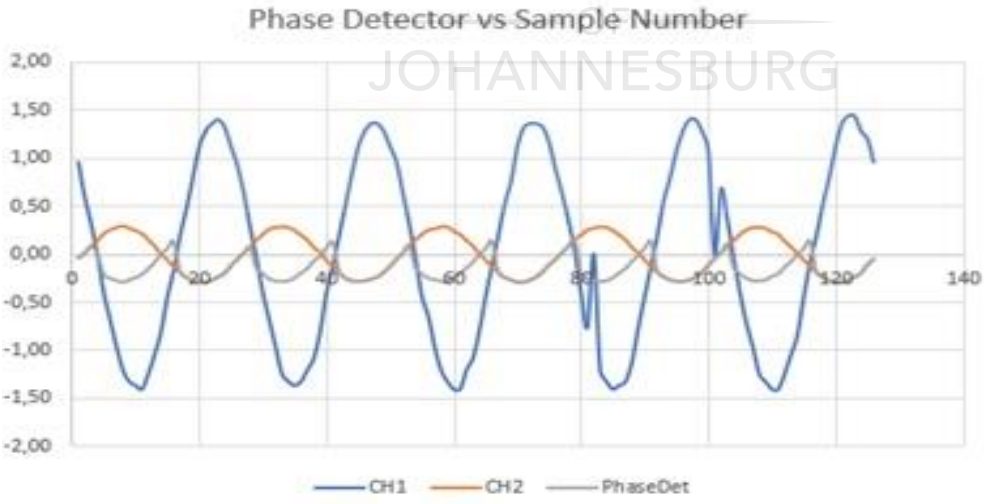


Figure 11. *Phase error generated by the new modified Costas phase detector with memory consideration*

The next step in the experimental setup was to pass the confirmed complex signal to the I/Q demodulator to recover I and Q.

4.5. Summary

In this chapter the results of two experiments on signal data collection and analysis of signal data were presented and discussed. The first experiment was to develop a method of collecting digitized signal data. The second experiment used the collected signal data for processing further. Spectral results for experiment two has shown that sample frequency is reduced.

Chapter 5: Case Study 2: Experimental Evaluation of Step-Size and Dynamic Range

5.1. Introduction

This Chapter develops simulation model of odd/even order sampling. The model is used to evaluate the performance of state-of-the-art sampling method to traditional sampling methods. Analog-to-digital sampling parameters such as step-size and dynamic range were investigated.

5.2. Overview of Experimental Setup

In chapter 3 the mathematical model for odd/even order sampling with internal tuning of sampling in such a way a selection between odd and even sampling is achieved with ease. In this section a simulation model that aims to compare performance of the odd/even order sampling model to 1st order delta-sigma model is outlined.

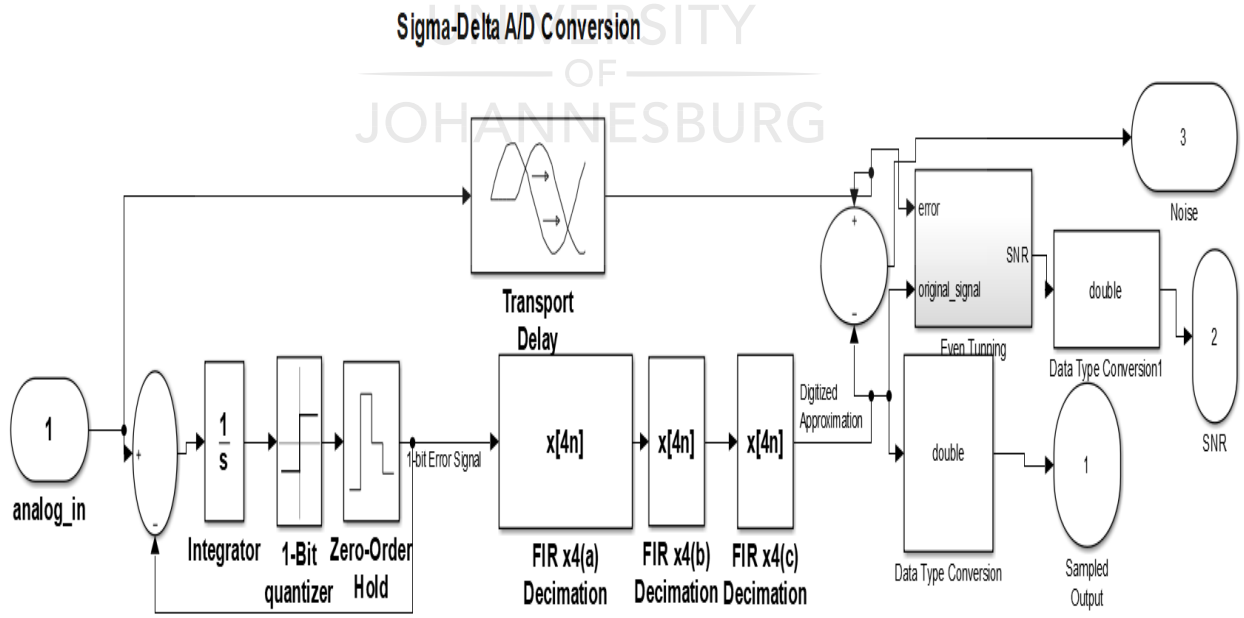


Figure 12. *Simulink experimental setup design for 1st order delta-sigma model*

The delta-sigma model used in the proposed simulation experiment consists of a negative feedback loop (delta part of the design), first order integrator (sigma or summation part of the design), 1-bit quantization (first stage for digital formation), zero-order-hold (sampling in discrete-time as second stage digital formation), three stages 4th order digital filtering to eliminate sampling noise (digital filtering as the third and final stage of the ADC model).

5.3. Experiment One – Analysis of Step-Size Behavior

Simulation results for delta-sigma sampling are shown in Figures 13 and 15 with quantization step size clearly defined. Quantization step size measurement attributes for the same delta-sigma sampling are given in Table 24. These results in Table 24 are as expected as they indicate that the sampling frequency is reduced from 8 kHz in delta-sigma sampling to 2 kHz in the new odd/even ordered sampling.

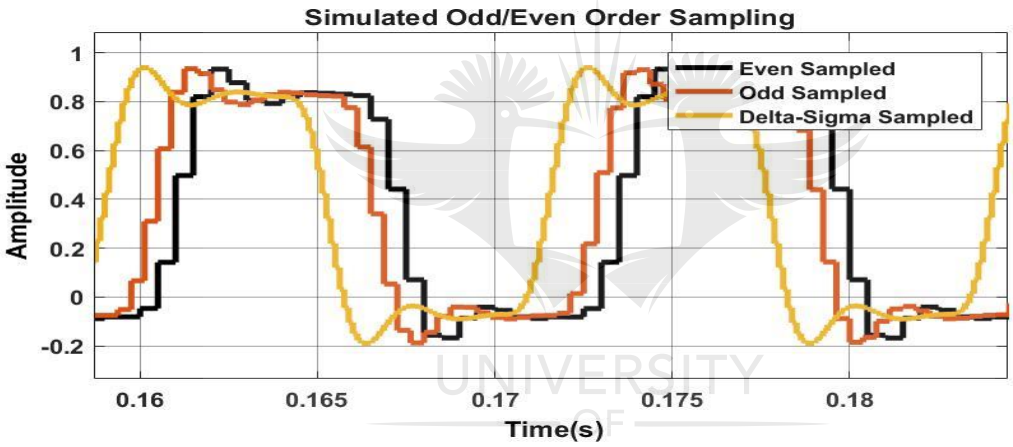


Figure 13. Simulated odd/even order sampling

Figure 13 above portrays the phase performance between 1st order delta-sigma and odd/even order sampling. The phase difference between 1st order delta-sigma and odd/even sampling was created on design by the transport delay distinguish between the signals and should be ignored for analysis. 1st delta-sigma is indicated by the yellow, odd order sampling by red and even order sampling by black. It should be noted there is (n-1) phase lag between even order sampling (black) and odd order sampling (red).

Ordered sampling adapted 4^{th} order designed in Figure 14, the design retains the negative feedback, first order integrator, 1 – bit quantization and decimation. Ordered samples selector is introduced as shown in Figure 14.

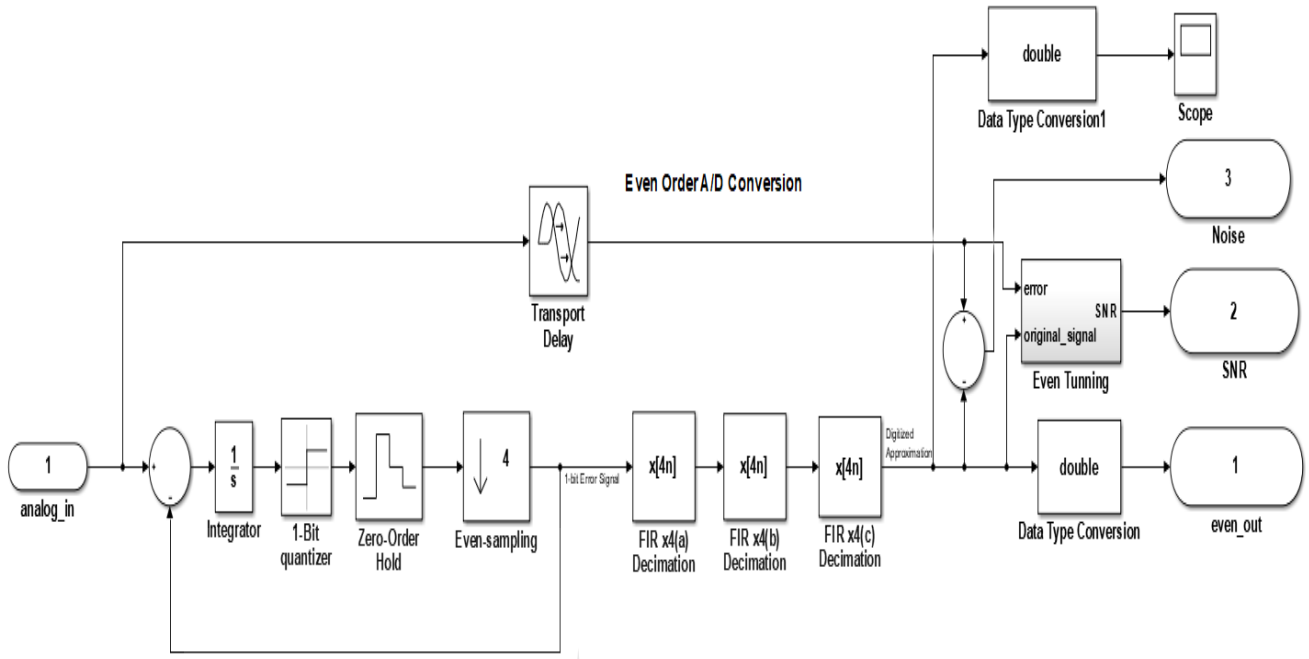


Figure 14. *Simulink experimental setup design for ordered sampling derived from 1st order delta-sigma model.*

Table 24. *1st order delta-sigma and proposed ordered simulation measurements results.*

Quantization Step Parameter	Sim Results (Δ-∑)	Sim Results (Proposed)
ΔT	124.687us	378.747us
ΔY	0.1477 volts	0.4120 volts
ΔF	8.020 kHz	2.640 kHz
ΔY/ΔT	1.185 (Volts/ms)	1.088 (Volts/ms)

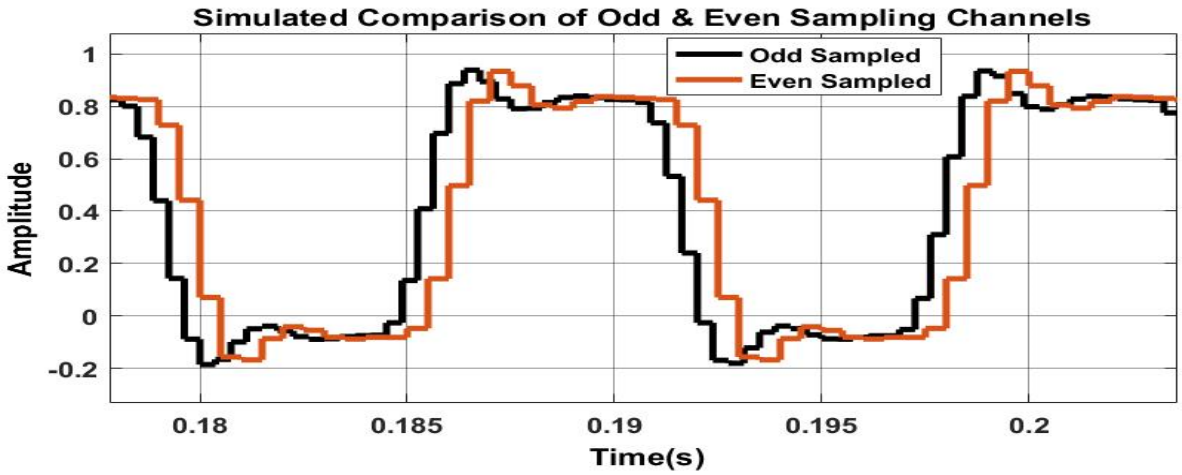


Figure 15. *This portrays the phase lag performance between odd and even order sampling. Odd order sampling marked by black and even order sampling by red. It should be noted the (n-1) phase lag between even order sampling (red) and odd order sampling (black) varies per period interval*

5.4. Experiment Two – Analysis of Dynamic Range Behaviour

During the simulation investigation we also undertake mean square error (MSE) performance evaluation of the dynamic range for the proposed new odd/even ordered sampling. MSE performance between odd and even components of the ordered sampling was the system under investigation with 1st order delta-sigma being assumed to be the input signal with minimum distortion. To confirm the effects of ordered sampling on dynamic range distortion the even ordered sampling was made to be 4th order while the odd ordered sampling was made to be 1st order as shown in Table 25.

Table 25. *Dynamic range behaviour simulation experimental setup parameters, with the order of even ordered sampling significantly increased in comparison with odd ordered sampling*

Ordered Sampling	Order
Odd	1^{st}
Even	4^{th}

The results indicate that the dynamic range distortion increases by 0.1 Volts/ms for every increasing order as shown in MSE results in Figure 16 below.

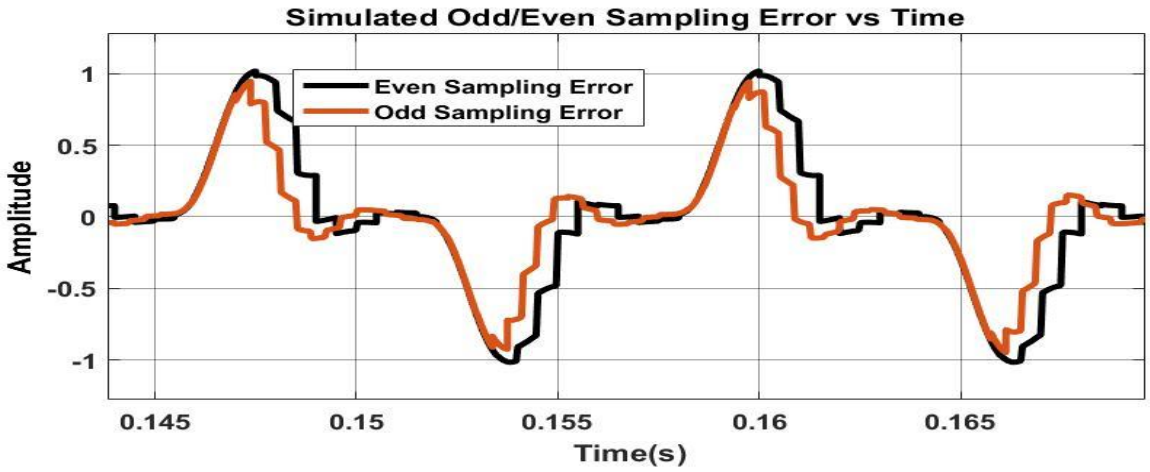


Figure 16. *Ordered sampling dynamic range investigation results with proposed new odd/even ordered sampling inter-component investigation undertaken in Simulink*

5.5. Summary

In this chapter the results of two experiments on analysis of step-size behaviour and analysis of dynamic range behaviour were presented and discussed. The first experiment was to investigate step-size quantization. The second experiment was to investigate dynamic range. Results have shown that step-size quantization error increases with increasing order as shown in Table 16, while the dynamic range is not significantly affected.

Chapter 6: Case Study 3: Signal-To-Noise Ratio (SNR) Verification of Superiority of Ordered Analog-To-Digital Sampling

6.1. Introduction

This chapter proposes new soft-core architecture for odd/even order sampling with memory considerations shown in Figure 17. The architecture uses digital signal processing (DSP) slices under the intelligent Processing sub block. Block-RAM store signal data, Registers interface with Scalar Processing sub block and Process Containers contain logic to configure both Block-RAM and Register configuration. Axi bus access is primary self-controlled but state transitions such as write, read and enable are externally controlled by Process Controllers. The derived equations 12 and 13 are implemented onto a DSP and signal data is loaded through the axi bus from Block-RAM. Different signal source data selection that switches between odd or even order through memory strobe and any other interesting application procedure is software programmed through the Scalar Processing sub-block. This section will implement the proposed architecture in Figure 17 and Table 26 and compare results to simulation for practical performance evaluation.

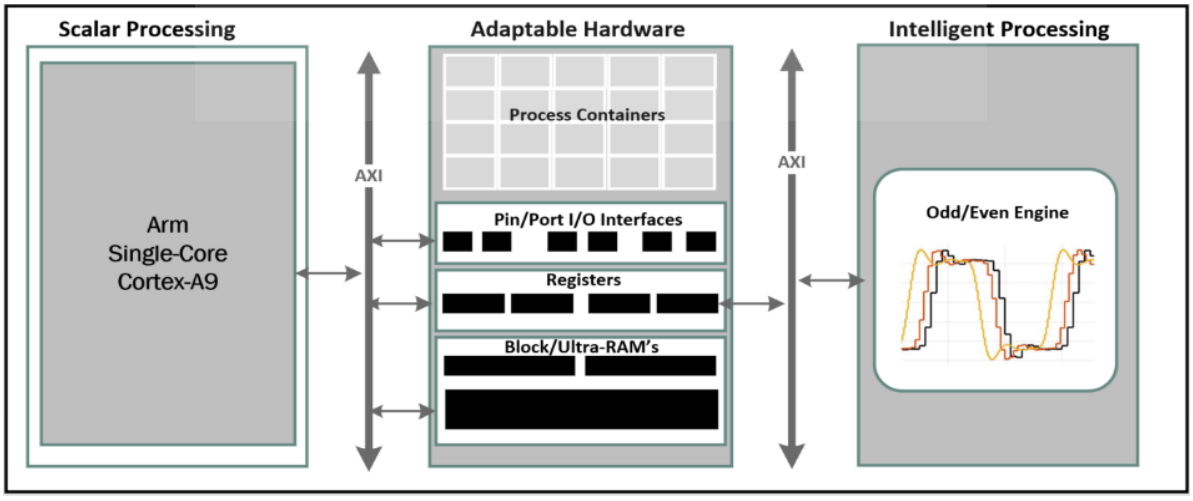


Figure 17. This presents the proposed new odd/even order sampling architecture block diagram

Table 26. Proposed new odd/even order sampling architecture breakdown

Sub Block	Functionality
Scalar Processing	**Access to Registers as Datatype for Software**
Adaptable Hardware	**Process Container access to Digital Logic(LUT), Pin/Ports, Registers, Block RAM**
Intelligent Processing	**Access to a ground of DSP Slices for equation manipulation**

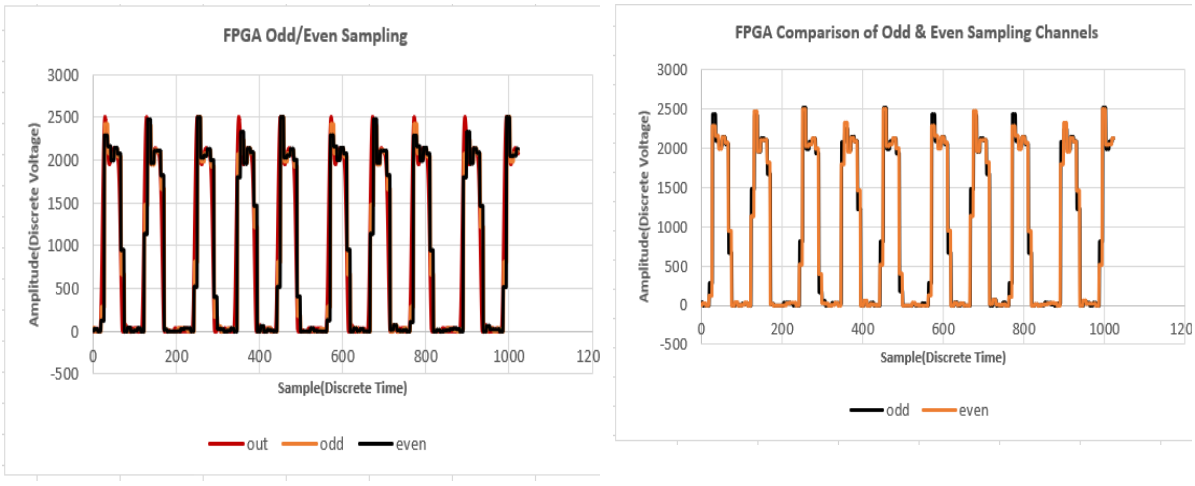
6.2. Experiment One – Verification of Practical Application of New Odd/Even Order Sampling

Laboratory verified signals captured in Chapter 4 were loaded and stored into Xilinx FPGA flash memory to accommodate implementation into FPGA chipset. Resource utilization for the proposed new soft-core architecture for odd/even order sampling is presented in Table 27 shown below with total resource utilization sitting just below 50%.

Table 27. FPGA resource utilization for the new odd/even order sampling

	Used	Available	Utilization
SliceUtilization			
Slice LUTs	2044	14400	14.19%
LUT as Logic	1828	14400	12.69%
LUT as Memory	216	6000	3.60%
SliceRegUtilization			
Reg as Flip Flop	3158	28800	10.97%
Reg as Latch	0	28800	0.00%
MultiplexerUtilization			
F7 Muxes	52	8800	0.59%
F8 Muxes	5	4400	0.11%
MemoryUtilization			
Black RAM	1.5	50	3.00%
DSPUtilization			
DSPs	2	66	3.03%
SpecificFeatureUtilization			
XADC	0	1	0.00%
Total Utilization			**48.15%**

Step size performance shown in Figures18(a) and 18(b) is comparable to the expected results derived from simulation presented in Chapter 4. The error performance in Figure 19 is also comparable with expected simulation results in chapter 4.



(**a**) (**b**)

Figure 18. *FPGA architecture implementation results: (a) Source signal which is in phase with I and odd/even order sampling results; (b) Odd and even order sampling results with Vivado integrated logic analyzer (ILA) window configured to 1 kilobytes*

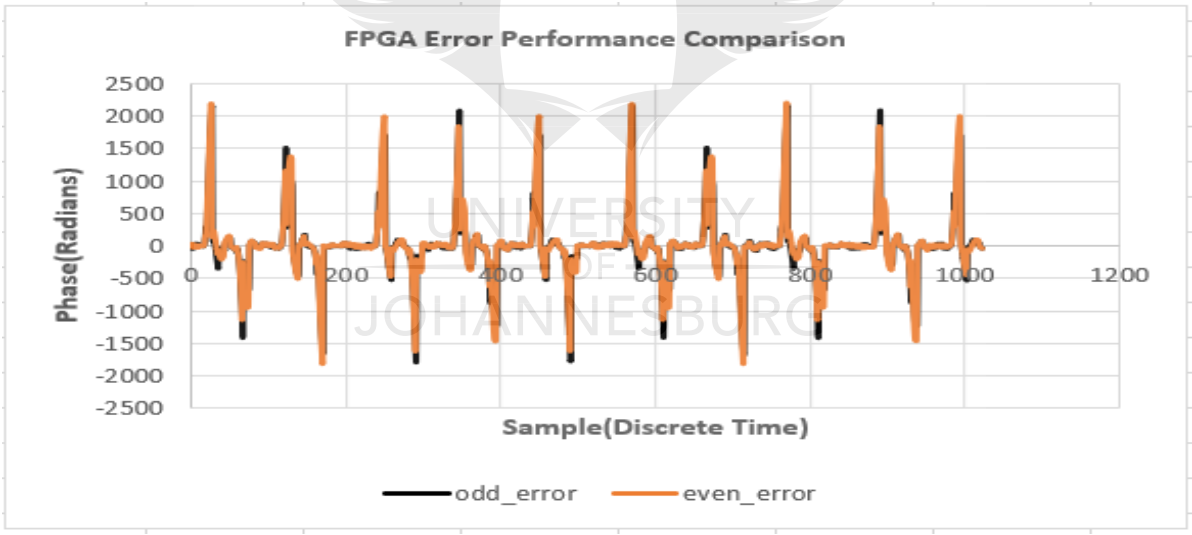


Figure 19. This figure presents error performance for odd/even order sampling

6.3. Experiment Two – Signal-To-Noise Ratio (Snr) Performance Comparison to Literature Available Sampling Schemes

Implementation from the previous subsection was followed by implementation of other sampling schemes from literature such as Mod-Δ, Mod-Δ (Gaussian) and Mod-Δ (Sinusoidal). According to literature [13] Mod-Δ(Gaussian) is a gaussian process whose PSD is flat within the designed band and Mod-Δ(Sinusoidal) is a sinusoidal waveform whose frequency is chosen at random, uniformly on [0, B), and whose amplitude is square root of covariance.

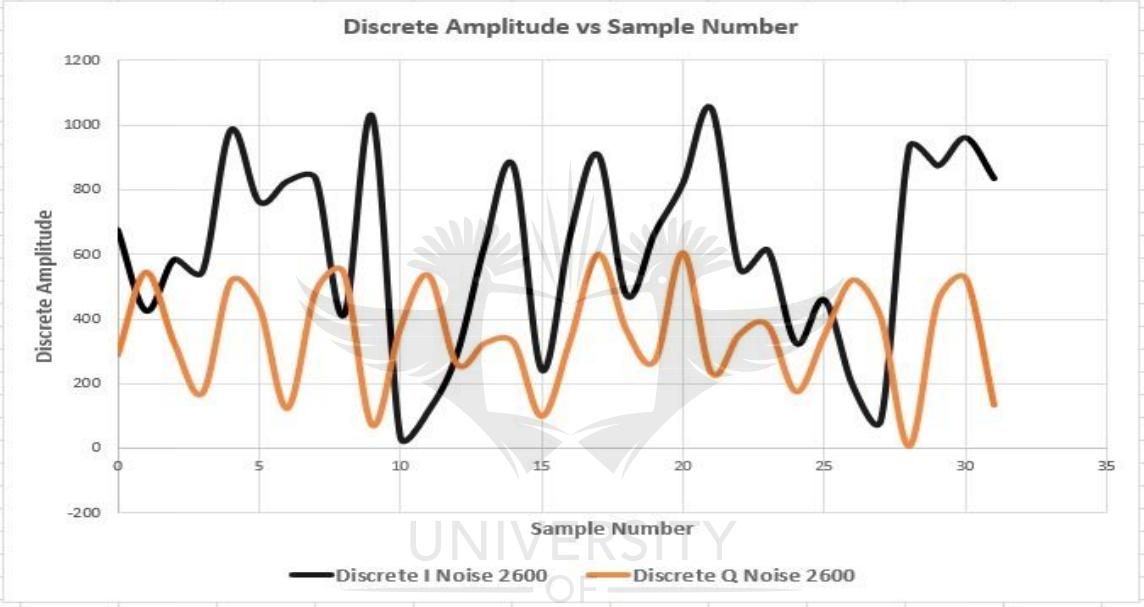


Figure 20. *This figure presents noise captured from Radar source on a field test.*

Many field tests were conducted to investigate realistic noise behaviour of Radar systems. The field tests were conducted at OR Tambo International Airport. The best noise performance was stored, and noise extracted for both I and Q components as shown in Figure 20. In Mod-Δ (Sinusoidal) the sinusoidal can be replaced by noise in the Figure 20 to make Mod-Δ (Noise) as it is within the specification of random frequency.

Table 28. FPGA resource utilization for the existing Mod-Δ sampling

	Used	Available	Utilization
SliceUtilization			
Slice LUTs	2044	14400	14.19%
LUT as Logic	1828	14400	12.69%
LUT as Memory	216	6000	3.60%
SliceRegUtilization			
Reg as Flip Flop	3158	28800	10.97%
Reg as Latch	0	28800	0.00%
MultiplexerUtilization			
F7 Muxes	52	8800	0.59%
F8 Muxes	5	4400	0.11%
MemoryUtilization			
Black RAM	1.5	50	3.00%
DSPUtilization			
DSPs	20	66	3030%
SpecificFeatureUtilization			
XADC	0	1	0.00%
Total Utilization			**75.45%**

Mod-Δ sampling was implemented under Intelligent Processing sub block while other components of architectural design in Figure 17 and Table 26. The mathematical model equations of the new odd/even order sampling were replaced by that of Mod-Δ sampling. The modification is also evident resource utilization in Table 28 registering DSP usage of just over 30% with total resource utilization sitting just above 75%.

Table 29. FPGA resource utilization for the existing Mod-Δ (Gaussian) sampling.

	Used	Available	Utilization
SliceUtilization			
Slice LUTs	2044	14400	14.19%
LUT as Logic	1828	14400	12.69%
LUT as Memory	216	6000	3.60%
SliceRegUtilization			
Reg as Flip Flop	3158	28800	10.97%
Reg as Latch	0	28800	0.00%
MultiplexerUtilization			
F7 Muxes	52	8800	0.59%
F8 Muxes	5	4400	0.11%
MemoryUtilization			
Black RAM	1.5	50	3.00%
DSPUtilization			
DSPs	22	66	33.33%
SpecificFeatureUtilization			
XADC	0	1	0.00%
Total Utilization			**78.48%**

Implementation Mod-Δ sampling was implemented under Intelligent Processing sub block was followed by that of Mod-Δ (Gaussian). The mathematical model equations of Mod-Δ sampling were replaced by that of Mod-Δ (Gaussian) sampling. The modification is also evident in resource utilization in Table 29 registering DSP usage of just over 33% with total resource utilization sitting just above 78%.

Table 30. FPGA resource utilization for the existing Mod-Δ (Noise) sampling

	Used	Available	Utilization
SliceUtilization			
Slice LUTs	2044	14400	14.19%
LUT as Logic	1828	14400	12.69%
LUT as Memory	216	6000	3.60%
SliceRegUtilization			
Reg as Flip Flop	3158	28800	10.97%
Reg as Latch	0	28800	0.00%
MultiplexerUtilization			
F7 Muxes	52	8800	0.59%
F8 Muxes	5	4400	0.11%
MemoryUtilization			
Black RAM	1.5	50	3.00%
DSPUtilization			
DSPs	27	66	40.91%
SpecificFeatureUtilization			
XADC	0	1	0.00%
Total Utilization			**86.06%**

Implementation Mod-Δ (Gaussian) sampling was implemented under the Intelligent Processing sub block was followed by that of Mod-Δ (Noise). The mathematical model equations of Mod-Δ (Gaussian) sampling were replaced by that of Mod-Δ (Noise) sampling. The modification is also evident resource utilization in Table 30 registering DSP usage of just over 40% with total resource utilization sitting just above 86%.

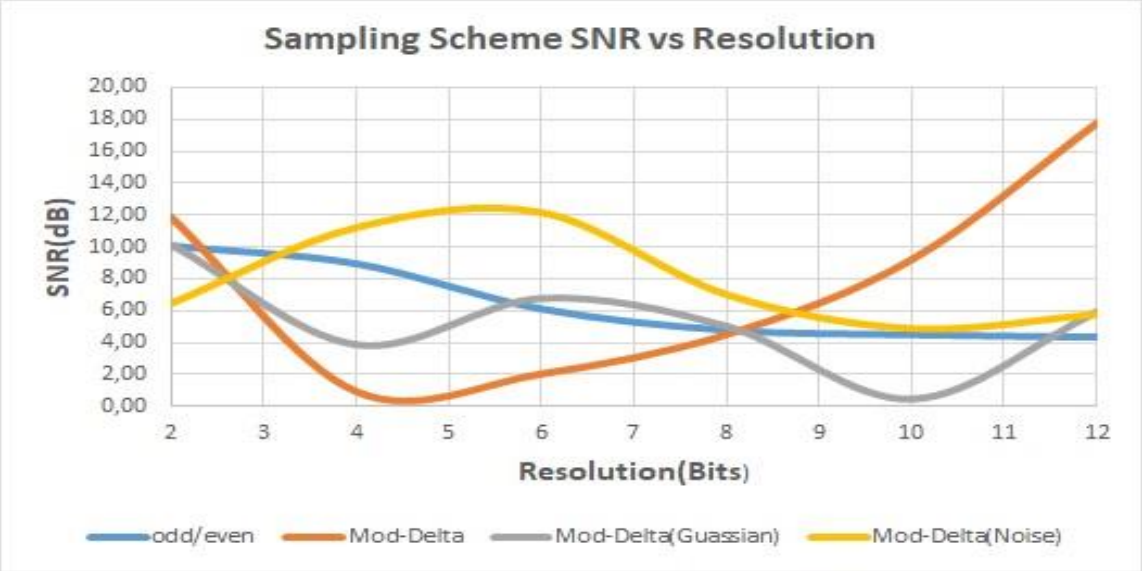


Figure 21. *This figure presents SNR performance between the new odd/even order, Mod-Δ, Mod-Δ(Gaussian) and Mod-Δ(Noise) sampling*

After the four implementations were usable SNR investigation was conducted on Vivado ILA tool. The investigation was through experiments for quantization resolution from 2 to 12 bits and SNR results are presented in Figure 21. SNR show that below 3 bits resolution Mod-Δ has the worst performance registering 12 dB signal distortion, Mod-Δ(Noise) has worst overall performances for all types of quantization designs between 2 and 12 bits while the new odd/even has average overall performance. Above 8 bits the new odd/even order sampling and Mod-Δ (Gaussian) give the best performance with the new odd/even order sampling recording 6 dB

6.4. Summary

In this chapter the results of two experiments on verification of practical implementation of odd/even order sampling and signal-to-noise ratio comparison to sampling schemes from literature were presented and discussed. The first experiment was to investigate and verify the practical implementation of odd/even order sampling on the MiniZed development board. The second experiment was to investigate signal-to-noise ratio in relation to existing sampling schemes. Results have shown that odd/even order sampling has the best performance at above 11 bits resolution.

Chapter 7: Case Study 4: Practical Implementation of Noise Jamming

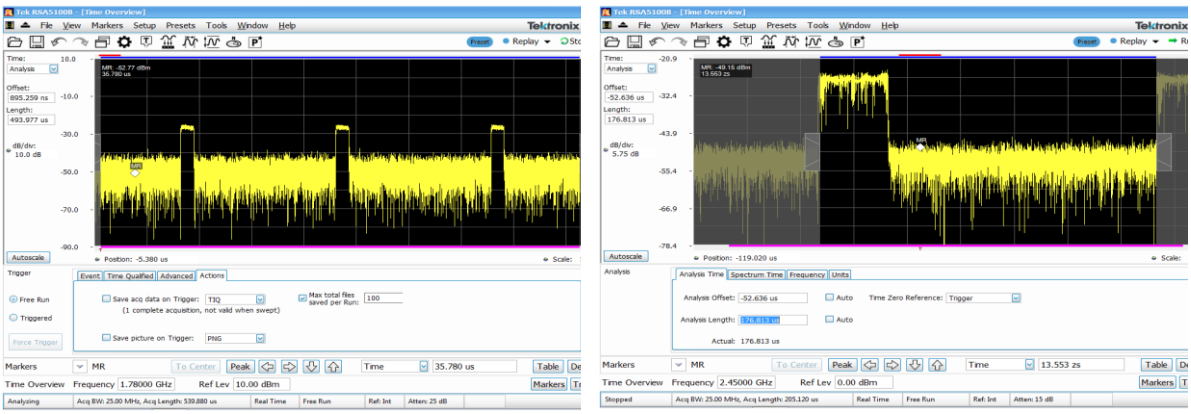
7.1. Introduction

This chapter investigates the signal-to-noise ratio performance trade-offs between field gathered noise from pulse Doppler radar, and that of simulation built from a gaussian modelled noise. Field experiments were conducted at OR Tambo International Airport and pulse Radar with pulse repetition interval of 176.81us was selected as the device under investigation. The investigation is embodied in two experiments in the next sub chapters.

7.2. Experiment One – Field Based Noise Data Collection

This work was developed in two approaches, the first was an effort to acquire realistic noise data from industrial Radar emitters. The aim was to match existing levels in Radar devices, and the facility with world class Radar systems that operate daily is OR Tambo International. OR Tambo International monitors the airspace using such Radars for air control reasons. Field studies were conducted with a spectral analyzer and different waveforms from coherent and non-coherent Radars. Two good quality non-coherent Radar waveforms were used for noise extraction. During noise extraction, 32 samples of waveforms were separated and stored as new noise waveforms. The 32 sampled waveforms were stored in FPGA flash memory and read over a defined clock cycle to reproduce the noise.

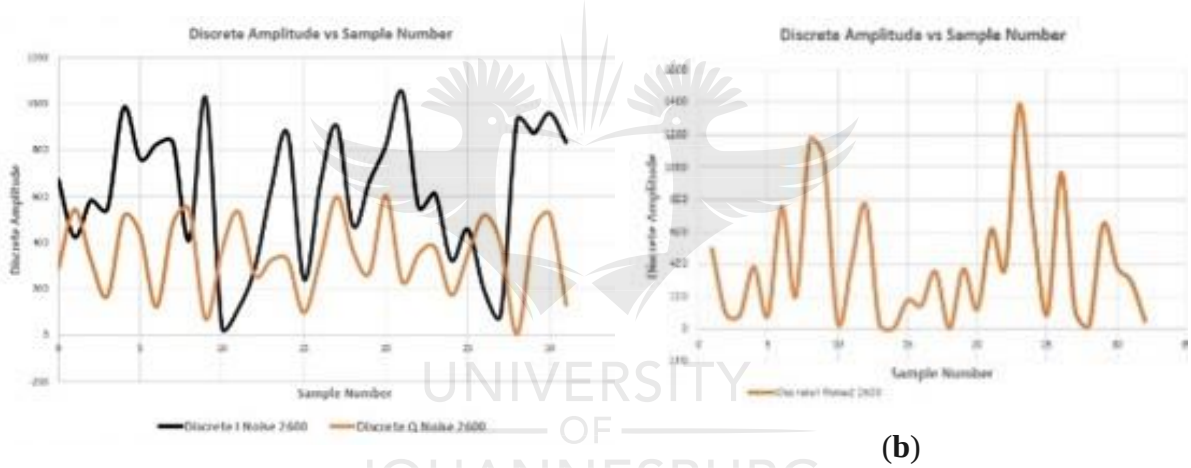
The second approach was conducted on the FPGA platform, gaussian noise was implemented using the DSP slices of the FPGA. Three waveforms were evaluated using Integrated Logic Analyser (ILA), one for gaussian noise, second for noise 1 and the third for noise 2. Additive noise (for gaussian, noise 1 and 2 respectively) function was applied to the input signal from Radar synthesizer to achieve noise jamming. Post analysis includes signal-to-noise ratio equations to conduct performance comparison between gaussian and field acquired noise. This subsection details the experimental investigations and results, experiment one dives straight to the experimental results of noise generation and experiment two built up on noise generation to achieve noise jamming. Field studies were conducted in OR Tambo International and several Radar performances were evaluated to determine the results using the spectral analyzer; the results are presented in Figures 22a to 22b.



(**a**) (**b**)

Figure 22. *Field test Pulse Radar (a) time behaviour of the Radar under test and (b) pulse repetition interval of the Radar*

Noise from Figures 22a and 22b were separated and only 32 samples were recorded and saved towards the noise generation as shown in Figures 23a and 23b.

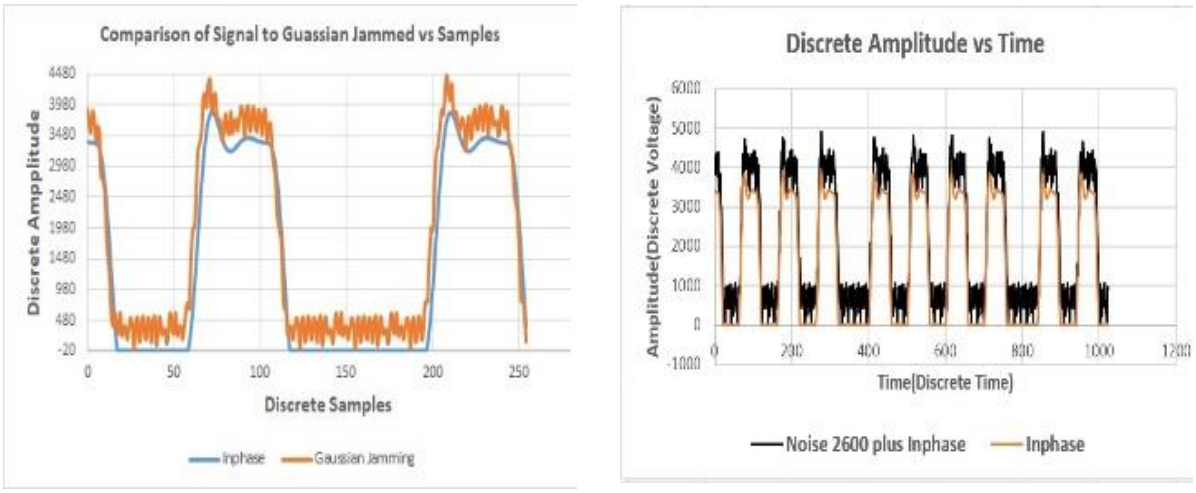


(**a**) (**b**)

Figure 23. *Separated noise behaviour from the Radar under test with (a) being noise 1 and (b) being noise 2*

7.3. Experiment Two – Noise Jamming

Noise jamming of two schemes was implemented on the FPGA platform, one in Figure 24(a) is theoretical gaussian noise generated using the gaussian equation. The second scheme in Figure 24(b) is the proposed noise jamming from realistic noise from Radar sources.



(**a**) (**b**)

Figure 24. Noise jamming results for (a) Gaussian noise jamming and (b) proposed noise jamming

FPGA resources were planned and optimized on the Xilinx Vivado platform. Our best efforts in implementing a noise jamming system capable of running hardware for both gaussian and proposed noise jamming is shown in Table 3. In should be noted that the major usage was on basic logic slices with only 1 DSP used for both addition operations of gaussian plus additive noise.

Table 31. *FPGA resource planning and utilization*

	Used	Available	Utilization
SliceUtilization			
Slice LUTs	2765	14400	19.20%
LUT as Logic	2522	14400	17.51%
LUT as Memory	243	6000	4.05%
SliceRegUtilization			
Reg as Flip Flop	4518	28800	15.69%
Reg as Latch	0	28800	0.00%
MultiplexerUtilization			
F7 Muxes	97	8800	1.10%
F8 Muxes	0	4400	0.00%
MemoryUtilization			
Black RAM	3	50	6.00%
DSPUtilization			
DSPs	1	66	6.00%
SpecificFeatureUtilization			
XADC	0	1	0.00%
Total Utilization			**69.55%**

Performance evaluation of the two schemes was done using SNR; the goal of this study is to achieve noise jamming with minimum effective noise power. Figure 25 shows that the proposed noise jamming schemes outperforms and theoretical gaussian jamming while achieving effective jamming SNR of 11 dB.

Figure 25. Signal to noise-ratio (SNR) performance comparison of the realistic and theoretical schemes

7.4. Summary

In this chapter the results of two experiments on field-based noise data collection and noise jamming were presented and discussed. The first experiment was to investigate time based behaviour of the field-based pulse Radar, parameters such as pulse repetition period and noise level were measured. The second experiment was to investigate implementation tradeoffs between noise captured from field activities and noise model by Gaussian distribution. Results have shown that noise jamming from Gaussian distribution has superior signal-to-noise ratio when compared to field-gathered noise.

Chapter 8: Conclusions and Recommendations

8.1. Conclusions

Chapter 3 presented numerical and experimental merits. Derived numerical demonstration of the proposed odd/even order sampling. Chapter 3 also used the numerical model to develop the simulation model to investigate MSE performance on odd/even order sampling. A clear description of numerical parameter that represents quantization error. Quantization error is very important in this study as it is largely influenced by the arrangements of samples. Figure 3 from simulation and Figure 10 from FPGA implementation seems to indicate that quantization error behaviour is consistent between simulation and implementation.

Chapter 4 develops an experimental setup to capture source signal data and used it to investigate the sample frequency performance using Excel for implementation of the numerical model. Spectral analysis on Excel in Figure 8 indicated that the sample frequency has reduced after odd/even order sampling; this validates the research assumption that initiated this study. Chapter 4 also implemented the proposed odd/even order sampling, Mod-Δ, Mod-Δ (Gaussian) and Mod-Δ (Noise) on an FPGA platform to capture computational requirements. Chapter 4 went further to investigate SNR performance for the four sampling schemes to demonstrate merits that the new odd/even order sampling has.

It is important to note that the quantization error based on MSE calculation for both simulation and FPGA implementation is comparable as shown in Figure 16 for simulation and Figure 19 for implementation. A detailed step size and dynamic range investigation revealed that the step size reduced from 8 kHz to 2 kHz, while the dynamic range slightly reduced from 1.185 Volts/ms to 1.088 Volts/ms. Implementation investigations have revealed that FPGA resource utilization for the new odd/even order sampling, Mod-Δ, Mod-Δ (Gaussian) and Mod-Δ (Noise) respectively is approximately 45%, 75%, 78% and 86%. This is a clear indication that is very lightweight when it comes to computation requirements as it has the least resource utilization as compared to the studied schemes.

By reducing the sample frequency of digitized signal, the issue of complicated architecture and expensive FPGA selection can be reduced. This paper proposes a novel odd/even order

sampling architecture that eliminates odd and even sampling on quadrature signals to reduce sample frequency. Realistic case studies that investigate the behaviour of the proposed new novel odd/even order sampling architecture, using computational simulation, laboratory and implementation experimentation.

- Simulation investigated step-size, dynamic range and dynamic range error behaviour. Results verify that odd/even ordered sampling can significantly reduce the sample frequency from 8 kHz to 2 kHz while not adversely affecting the dynamic range.
- Laboratory experimentation investigated feasibility of research assumption that ordered sampling reduces sample frequency. Results verify this assumption using time and spectral analysis.
- Implementation experimentation investigated feasibility of implementing ordered sampling on FPGA platform in comparison to sampling architecture in literature. While on the other hand research investigated SNR behaviour of odd/even ordered sampling in comparison to Mod-Δ, Mod-Δ (Gaussian) and Mod-Δ (Noise) literature. Results indicate that odd/even order sampling is the most economical in comparison to architectures evaluated with resource utilization at 45%. SNR results were not conclusive for sampling resolution below 8 bits, for resolution between 8 bits and 11 bits odd/even ordered sampling is the second-best performer, while for sampling resolution above 11 bits performance is the best.

8.2. Recommendations for Further Work

Future studies should involve a detailed study of computation requirements of the different sampling schemes with numerical tracking of the cause for deviations in resource utilization. Mod-Δ (Sinusoidal) has two constraints, frequency domain and amplitude constraints defined as:

- Random frequency uniformly distributed (0, B)
- Amplitude is a square root of covariance

Mod-Δ (Sinusoidal) generally require the sinusoid to be noise to meet the frequency constraint while the amplitude of that noise must be a square root of covariance. The sinusoid used in this paper was a noise signal acquired realistic Radar equipment as shown in Figure 11, this acquisition was made without verification conducted against Mod-Δ (Sinusoidal) amplitude

constraint. Future work should also include amplitude investigation into the implementation of Mod-Δ (Noise) that validates if it satisfies the square root of covariance constraint.

Further study of Modified-Costas algorithm towards electronic warfare applications, and investigation into performance of quantization error should be undertaken as well as further study of peak detection algorithm towards electronic warfare, investigation into performance of dynamic range and signal-to-noise ratio as these usually affect amplitude performance. Digital resource utilization for both Modified-Costas and peak detection could also be investigated in comparison to existing phase and amplitude detection techniques.

References

[1] M. McGrath and P. Marshall, “ECM Techniques Generator,” Midwest Symposium on Circuits and Systems, pp. 1749 - 1752, 2005.

[2] R. Li, B. Slocumb and P. West, “Tracking in the Presence of Range Deception ECM and Clutter by Decomposition and Fusion,” in Proceedings of the SPIE, Conference on Signal and Data Processing of Small Targets, 1999.

[3] G. Lu, B. Tang and G. Gui, “Deception ECM signals cancellation processor with joint time-frequency pulse diversity,” in IEICE Electronics Express, 2011.

[4] G. Shixiang And W. Xizhang , “Noise Jamming To Isar Based On Multiplication Modulation,” In Iet International Radar Conference 2013, Xi'an, 2013.

[5] Z. Li, N. Tai, C. Wang and N. Yuan , “A Study on Blanket Noise Jamming to LFM Pulse Compression Radar,” in 2017 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Xiamen, 2018.

[6] Y. Xun, Z. Cui and X. Hao , “Optimal Parameters Selection of Noise FM Jamming and Its Simulation,” in 2011 International Conference on Consumer Electronics, Communications and Networks (CECNet), XianNing, 2011.

[7] H. Hong-xu, Z. Yi-yu, J. Wen-li and H. Zhi-tao, “A New Time-Delay Echo Jamming Style to SAR,” in 2010 2nd International Conference on Signal Processing Systems (ICSPS), Dalian, 2010.

[8] H.-x. Huang , Z.-t. Huang and Y.-y. Zhou, “Jamming Research to SAR based on Frequency Characteristic,” in 2010 2nd International Conference on Signal Processing Systems (ICSPS), Dalian, 2010.

[9] B. Lv, “Simulation Study of Noise Convolution Jamming Countering to SAR,” in 2010 International Conference On Computer Design And Appliations (ICCDA 2010) , Qinhuangdao, 2010.

[10] W. Ye, H. Ruan, S.-x. Zhang and L. Van , “Study of Noise Jamming Based on Convolution Modulation to SAR,” in 2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE), Changchun, 2010.

[11] H. Hong-xu, Y.-y. Zhou, J. Wu and Z.-t. Huang, “A Frequency-based Inter/Intra Partly Coherent Jamming Style to SAR,” in 2010 2nd International Conference on Signal Processing Systems (ICSPS), Dalian, 2010.

[12] S. Jun-jun, B. Da-ping and X. Lei , “Novel Evaluation Method of Jamming Effect on ISAR Based on Target Detection,” in 2009 2nd Asian-Pacific Conference on Synthetic Aperture Radar, Xian, Shanxi, 2009.

[13] D. Chunxi, Y. Shaoquan, Z. Guoqing and Z. Yu , “Effect of Noise FM Jamming against ISAR Imaging,” in Effect ofNoise FM Jamming against ISAR Imaging, Shanghai, 2006.

[14] P. Shi-rui and L. Xin , “Time Synchronization Error and PRF in Pulse Jamming Against ISAR,” in 2011 International Conference on Wireless Communications and Signal Processing , Nanjing, 2011.

[15] T. J. Moeller, Field Programmable Gate Arrays For Radar Front-End Digital Signal Processing, Massachesetts Institute Of Technology, 1999.

[16] A. Marzuki, A. M. Shakaff and Z. Sauli, “Design of I/Q Demodulator for W-CDMA,” in ICSE International Conference on Software Engineering, Johor Bahru, Malaysia, 2008.

[17] V. I. Slyusar and P. E. Serduk, “A Two Channels I/Q-Demodulator,” in Proceedings of International Conference on Modern Problem of Radio Engineering, Telecommunications and Computer Science, Lviv, Ukraine, 21-24 February 2012.

[18] E. Brunner, “An I/Q Demodulator with Phase Shifter for Beamforming Applications,” in IEEE Ultrasonics Symposium, Vancouver, Canada, 2006.

[19] R. Xue, K. F. Chang and K. W. Tam, “A New Method of an IF I/Q Demodulator for Narrowband Signals,” in IEEE International Symposium on Circuits and Systems, Kobe, Japan, 23-26 May 2005.

[20] R. M. Shibu and A. Gopalakrishnan, “High Performance Computing for Sound Wave Using Multi Channel Digital I/Q Demodulator,” in International Conference on Control Communication and Computing (ICCC), 2013.

[21] V. I. Slyusar, “I/Q-Demodulation Of The Odd Order,” in International Conference on Antenna Theory and Techniques,, Kharkiv, Ukraine, 2015.

[22] T. Boles, R. Houlihan and C. Weigand, “An HMIC I/Q Modulator/Demodulator for RFID Applications,” in Asia-Pacific Microwave Conference Proceedings, Suzhou, China, 2005.

[23] Y.-T. Chang, H.-Y. Chen and H.-C. Lu, “A V-Band High Linearity Sub-harmonic I-Q Demodulator Using Transformer Coupling,” in IEEE International Symposium on Radio-Frequency Integration Technology (RFIT), Seoul, Korea, 2017.

[24] J. Kobayashi, F. Ozawa, Y. Toriyama, T. Shirosaki, T. Taniguchi and A. Matsuzawa, “Characteristics of a 60 GHz direct conversion quadrature modulator/demodulator,” in The 7th German Microwave Conference, Germany, 12-14 March 2012.

[25] Y.-T. Chou, Y.-H. Lin and H. Wang, “A High Image Rejection E-Band Sub-Harmonic IQ Demodulator with Low Power Consumption in 90-nm CMOS Process,” in Proceedings of the 46th European Microwave Conference, London, 4-6 October 2016.

[26] D. H. Sim, “CMOS I/Q Demodulator using a High-isolation and Linear Mixer for 2GHz Operation,” in IEEE Radio Frequency Integrated Circuits Symposium, 2004.

[27] B. E. Priyanto, C. L. Law and Y. L. Guan, “Design & Implementation of All Digital I-Q Mokdulator and Demodulator for High Speed WLAN in FPGA,” Victoria, Canada, 2003.

[28] V. Slyusar and E. Zhivilo, “The Synthesys of Equivalence Digital Filters for Tandem Decimation on Base I/Q-Demodulation,” Kharkov, Ukraine, 2017.

[29] K. Ruman, A. Pietrikova, I. Vehec, T. Rovensky and P. Galajda, “Modified I-Q Demodulator for M-Sequence UWB Sensor System Based on LTCC,” in 38th International Spring Seminar on Electronics Technology, Eger, Hungary, 6-10 May 2015.

[30] S. Katwal, R. Nath and G. Murmu, “A Simple Kalman Channel Equalizer using Adaptive Algorithms for Time Variant Channel,” in Proceedings of 2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN 2011), Thuckalay, Tamil Nadu, India, 21-22 July 2011.

[31] S. D. Silverstein, “Blind Equalization of Phase Aberations in Coherent Imaging: Medical Ultrasound and SAR,” in Proceedings of the Tenth IEEE Workshop on Statistical Signal and Array Processing , Pocono Manor, PA, USA, 16-16 Aug. 2000.

[32] J. L. Finol and M. Buchhoz, “Design of an Inphase and Quadrature Phase and Amplitude Imbalance Compansation in Quadrature Receivers,” in Proceedings of the Fifth IEEE International Caracas Conference on Devices, Circuits and Systems, Dominican Republic, Nov.3-5, 2004.

[33] C.-L. Lu, “Impacts of I/Q Imbalance on QPSK-OFDM-QAM Detection,” in IEEE Transactions on Consumer Electronics, August, 1998.

[34] M. Lont, D. Milosevic, G. Dolmans and M. H. Roermund, “Implications of I/Q Imbalance, Phase Noise and noise Figure for SNR and BER of FSK Receivers,” in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, AUGUST 2013.

[35] R. Laksmanam and F. harris, “Joint I/Q Mixer and Filter Imbalance Compensation and Channel Equalization with Novel Preamble Design,” in 2016 24th European Signal Processing Conference (EUSIPCO), Budapest, Hungary, 2 September 2016.

[36] J. Gao, X. Zhu, H. Lin and A. K. Nandi, “Linear Least Squares CFO Estimation and Kalman Filtering based I/Q Imbalance Compensation in MIMO SC-FDE Systems,” in 2010 IEEE International Conference on Communications, Cape Town, South Africa, 01 July 2010.

[37] J.-E. Eklund and R. Arvidsson, “A Multiple Sampling, Single A/D Conversion Technique for I/Q Demodulation in CMOS,” in IEEE JOURNAL OF SOLID-STATE CIRCUITS, DECEMBER 1996.

[38] I. Mashkin, A. Nikolaev and N. Nikitin, “Assessing the Impact of Analog-to-digital Converter Resolution on the Phase Difference Measurement Error in a Digital Receiver,” in International Scientific & Technical Conference "Dynamics of Systems, Machanisms and Machines", Omsk, Russia, 13-15 November 2018.

[39] D. Budanov, M. Pilipko and D. Morozov, “Encoders for flash analog-to-digital converters,” in 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, Moscow, Russia, 29 Jan - 1 Feb 2018.

[40] A. Mohammadi and M. Chahardori, “A Low-power, Bootstrapped Sample and Hold Circuit with Extended Input Range for Analog-to-Digital Converters in CMOS 0.18 μm,” in 2018 15th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), Prague, Czech Republic, 2-5 July 2018.

[41] V. N. Ashanin and A. Korotkov, “Realisation of Integrating Analog-Digital Converter with Intermediate Time-Impulse Modulation Using NI Multisim,” in 2018 Moscow Workshop on Electronic and Networking Technologies (MWENT), Moscow, Russia, 14-16 March 2018.

[42] M. Abdelkader, A. Ali, A. Abdelaziz, M. W. Ismail, M. A. Refky and Y. Ismail, “A 200MS/s, 8-bit Time-based Analog to Digital Converter (TADC) in 65nm CMOS technology,” in 2016 Fourth International

Japan-Egypt Conference on Electronics, Communications and Computers (JEC-ECC), Cairo, Egypt, 31 May-2 June 2016.

[43] A. Glascott-Jones, M. Wingender, F. Bore, M. Stackler, N. Chantier, K. Salmi, P. Coquille, M. Martin, V. Monier, G. Wagner, E. Savasta, D. Bellin, R. Pilard and J. Duvernay , "Results from a Prototype 6GSps Digital to Analogue Converter with greater than 7 GHz Analogue Bandwidth.," in 2016 11th European Microwave Integrated Circuits Conference (EuMIC), London, UK, 3-4 Oct. 2016.

[44] T. Waho, "Analog-to-Digital Converters Using Not Multi-level But Multi-bit Feedback Paths," in 2017 IEEE 47th International Symposium on Multiple-Valued Logic (ISMVL), Novi Sad, Serbia, 22-24 May 2017.

[45] M.-K. JEON, W.-J. YOO, C.-G. KIM and C. YOO (Member, IEEE), "A Stochastic Flash Analog-to-Digital Converter Linearized by Reference Swapping," in 2017 IEEE. Translations and content mining are permitted for academic research, 26 October 2017.

[46] S. Jia, . L. Weng, W. Wang and Y. Wang, "A highly linear 5GS/s voltage-to-time converter for time-based analog-to-digital converters," in 2017 IEEE Asia Pacific Microwave Conference (APMC), Kuala Lumpar, Malaysia, 11 January 2018.

[47] P. Dhage and P. Jadhav , "Design of Power Efficient Hybrid FlashSuccessive Approximation Register Analog to Digital Converter," in 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 08 February 2018.

[48] O. Ordentlich, G. Tabak, P. K. Hanumolu, A. C. Singer and G. W. Wornell, "A Modula-Based Architecture for Analog-to-Digital Conversion," IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, vol. 12, no. 5, pp. 825 - 840, October 2018.

[49] D. Behera,, R. S. Kumar and N. Krishnapura, Member, IEEE, "Reset-Free Memoryless Delta–Sigma Analog-to-Digital Conversion," in 2018 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS–I, NOVEMBER 2018.

[50] X. Ding, K. Hofmann, L. Zhang, D. Yi and Y. Ma, "Redundant Double Conversion based Digital Background Calibration of SAR ADC with Convergence Acceleration and Assistance," in Proceedings of the 25th International Conference "Mixed Design of Integrated Circuits and Systems", Gdynia, Poland, June 21-23, 2018.

[51] P. Cruz, T. Alves and A. Cartaxo, "Relaxing the ADC Sampling Rate in High-Resolution Radar Systems Through Photonic Analogue-to-Digital Conversion," in 2018 20th International Conference on Transparent Optical Networks (ICTON), Bucharest, Romania, 1-5 July 2018.

[52] D. Budanov, D. Morozov and M. Pilipko, "An 8-bit Analog-to-Digital Converter With a Novel Encoder Using 90 nm CMOS," in 2018 IEEE International Conference on Electrical Engineering and Photonics (EExPolytech), St. Petersburg, Russia, 22-23 Oct. 2018.

[53] P. Yang, Y.-J. Cen, Y.-K. Li and D.-G. Li, "The Design of High Resolution Capacitance Array in SAR ADC," in 2018 IEEE 18th International Conference on Communication Technology (ICCT), Chongqing, China, 8-11 Oct. 2018.

[54] M. Pilipko and M. Manokhin, “Design of a Low-Power 12-bit SAR ADC,” in 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Saint Petersburg and Moscow, Russia, Russia, 28-31 Jan. 2019.

[55] B. Rahrovi, “A Review and Comparison of the DSP-Based Resolver to Digital Conversion Methods,” in 2019 IEEE Texas Power and Energy Conference (TPEC), College Station, TX, USA, USA, 7-8 Feb. 2019.

[56] X. GUO, G. MEHDI, M. ASIF, A. HU and J. MIAO, “1-Bit/2-Level Analog-to-Digital Conversion Based on Comparator and FPGA for Aperture Synthesis Passive Millimeter-Wave Imager,” IEEE Access, vol. 7, pp. 51933 - 51939, 12 April 2019.

[57] M. A. Bensenouci, H. Escid and M. Attari, “Neural network-based non-linear A/D conversion,” in 2015 IEEE International Circuits and Systems Symposium (ICSyS), Langkawi, Malaysia, 28 January 2016.

[58] H. Xu, L. Wang, R. Yuan and Y. Chang, “A/D converter background calibration algorithm based on neural network,” in 2018 International Conference on Electronics Technology (ICET), Chengdu, China, 23-27 May 2018.

[59] G. Vougioukas, P. N. Alevizos and A. Bletsas, “Coherent Detector for Pseudo-FSK Backscatter Under Ambient Constant Envelope Illumination,” in 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 27 August 2018.

[60] X. Ma, Student Member, IEEE,, L. Li, Member, IEEE,, S. Ming, X. F. I. You and J. Lin , Fellow, IEEE, “Envelope Detection for an ADC-Relaxed Double-Sideband Low-IF CW Doppler Radar,” IEEE Transactions on Microwave Theory and Techniques, vol. 66, no. 12, pp. 5833 - 5841, 20 September 2018.

[61] A. Golan, A. Etinger, M. Einat and Y. Pinhasi, Member, IEEE, “Digital Signal Detection by a Glow Discharge Detector,” IEEE Transactions on Plasma Science, vol. 47, no. 1, pp. 95 - 99, Jan. 2019.

[62] B. Thijssen, Student Member, IEEE,, E. Klumperink, Senior Member, IEEE, P. Quinlan, Member, IEEE and B. Nauta, Fellow, IEEE, “Feedforward Phase Noise Cancellation Exploiting a Sub-Sampling Phase Detector,” IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 65, no. 11, pp. 1574 - 1578, Nov. 2018.

[63] J. Mitra , Student Member, IEEE, and T. K. Nayak, “An FPGA-Based Phase Measurement System,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 26, no. 1, pp. 133 - 142, Jan. 2018.

[64] I. L. Syllaios, “Hybrid ΔΣ Programmable Phase/Frequency Detector for IoT Chipsets,” in 2018 IEEE International Symposium on Circuits and Systems (ISCAS), Florence, Italy, 04 May 2018.

[65] B. Ga˛sowski, S. Hanasz, K. Czuba and Ł. Zembala, “Influence of step recovery diode DC bias on AM/P conversion in sampling phase detectors,” in 2018 22nd International Microwave and Radar Conference (MIKON), Poznan, Poland, 09 July 2018.

[66] J. Lin, Z. Song, N. Qi, Member, IEEE, W. Rhee , Senior Member, IEEE, Z. Wang, Fellow, IEEE and B. Chi , Member, IEEE, “A 77-GHz mixed-mode FMCW signal generator based on bang-bang phase detector,” in 2017 IEEE Asian Solid-State Circuits Conference (A-SSCC), Seoul, South Korea, 28 December 2017.

[67] E. G. Kacharmina, A. A. Timofeev, V. V. Chudnikov and B. I. Shakhtarin, “Pulse phase-frequency detectors in high-speed frequency synthesizers,” in 2018Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), Minsk, Belarus, 10 September 2018.

[68] J. Han and R. Chen , Senior Member, IEEE, “RF MEMS In-Line Type Phase Detector With Large Dynamic Range,” IEEE Electron Device Letters, vol. 40, no. 5, pp. 792 - 795, May 2019.

[69] M. Qinglin, Y. Meng, Y. Zhenya and F. Haihang, “An Empirical Envelope Estimation Algorithm,” in 2013 6th International Congress on Image and Signal Processing (CISP 2013), Hangzhou, China, 16-18 December 2013.

[70] C. Jarne, Simple empirical algorithm to obtain signal envelope in three steps, Buenos Aires, 2017.

[71] J. Liu, W. Liu, H. Wanga, T. Tao and J. Zhang, “A novel envelope extraction method for multichannel heart sounds signal detection,” International Conference on Computer Science and Information Technology, 2011.

[72] L. Zhong, X. Guo, A. Ji and X. Ding, “A Robust Envelope Extraction Algorithm for Cardiac Sound Signal Segmentation,” May 201

[73] A. Robel And X. Rodet, “Efficient Spectral Envelope Estimation And Its Application To Pitch Shifting And Envelope Preservation,” In Preceding Of The 8th International Conference On Digital Audio Effects, Madrid, 2005.

[74] T. Kusano, K. Yatabe And Y. Oikawa, “Envelope Estimation By Tangentially Constrained Spline,” In International Conference On Acoustics, Speech, And Signal Processing, Brighton, 2018.

[75] D. Schwarz, “SpectralEnvelopesinSound AnalysisandSynthesis,” Universitat Stuttgart Fakultat Informatik, 1998.

[76] A. Das, S. Dash, B. C. Babu and A. K. Sahoo, “A Novel Phase Detection System for Linear AllDigital Phase Locked Loop,” 2012.

[77] D. Hu, FPGA-BASED DIGITAL PHASE-LOCKED LOOP ANALYSIS AND IMPLEMENTATION, Urban, Illinois: University of Illinois at Urban-Champaing, 2011.

[78] J.-H. Chae, G.-M. Hong, J. Park, M. Kim, H. Ko, W.-Y. Shin, H. Chi, D.-K. Jeong and S. Kim, “A 1.74mW/GHz 0.11-2.5GHz Fast-Locking, JitterReducing, 180° Phase-Shift Digital DLL with a Window Phase Detector for LPDDR4 Memory Controllers,” in IEEE Asian Solid-State Circuits Conference, Xiamen, Fujian, China , November 9-11,2015 .

[79] I. Ali, S.-J. Oh, H. Abbasizadeh, B. S. Rikan, M. R. Rehman, D.-S. Lee and K.-Y. Lee, “An Ultra Low Power Fully Synthesizable Digital Phase and Frequency Detector for ADPLL Applications in 55 nm CMOS Technology,” Suwon.

[80] J. Jiang, Y. Wu, and H. Wang, “Analysis of Active Noise Jamming Against Synthetic Aperture Radar Ground Moving Target Indication,” in 8th International Congress on Image and Signal Processing(CISO 2015), 2015.

[81] L. Run-hua and D. Ke-qing, “Barrage Noise Jamming Suppression Methods for Airborne Phased Array Radar”.

[82] H. Paik, N. Sastry and I. Prabha, “Effectiveness of Noise Jamming with White Gaussian Noise and Phase Noise in Amplitude Comparison Monopulse Radar Receivers,” in International Conference on Electronics, Computing and Communication Technologies (CONECCT 2014), Bangalore, 2014.

[83] Y.-R. Jeong, H.-S. Jo and J.-G. Yook, “Noise-Jamming Effect as a Countermeasure Against TEMPST During Hgh-Speed Signaling,” in Transactions on Electromagnetic Compatibility, 2015.

[84] T. A. Gulliver, “A Moment Method for the Detection of Synchronisation Probes in Worst Case Noise and Multitone Jamming,” in IEEE, 1991.

[85] F. Shang, B.-q. Gao, G. Liu and X. Xiao, “Bit Error Performance Analysis of FH/MSK System in Different Multi-tone Noise,” in IEEE, 2005.

[86] H. Paik, N. N. Sastry and S. Prabha, “Effectiveness of Noise Jamming with White Gaussian Noise and Phase in Amplitude Comparison Monopulse Radar Receivers,” in IEEE CONECCT, 2014.

[87] M. Karaim, H. Elghamrawy, M. Tamazin and A. Noureldin, “Investigation of the Effects of White Gaussian Noise Jamming on Commercial GNSS Receivers,” in IEEE, 2017.

[88] H. Hao, D. Zeng and P. Ge, “Research on The Method of Smart Noise Jamming on Pulse Radar,” in Fifth International Conference on Instrumentation and Measurement, 2015.

[89] Y. Wang, D. Su and W. Chen, “Radio Noise Jamming Analysis on Communication System,” in IEEE, 2009.

Appendix A: Nove Odd/Even Order Sampling Data Analysis

Table 32. *Calculation of odd order sampling components up to 11th order*

#	I	Q	I(N+1)3	I(N+1)5	I(N+1)7	I(N+1)9	I(N+1)11
0	3374	3432	3384	3374	3384	3406	3406
1	3366	3422	3384	3374	3384	3406	3406
2	3362	3410	3384	3374	3384	3406	3406
3	3360	3397	3362	3374	3384	3406	3406
4	3356	3386	3362	3374	3384	3406	3406
5	3348	3376	3362	3348	3384	3406	3406
6	3321	3367	3348	3348	3321	3321	3140
7	3256	3362	3348	3348	3321	3321	3140
8	3140	3360	3348	3348	3321	3321	3140
9	2963	3357	3140	3348	3321	3321	3140
10	2716	3351	3140	2716	3321	3321	3140
11	2404	3331	3140	2716	3321	3321	3140
12	2040	3276	2404	2716	3321	3321	3140
13	1638	3175	2404	2716	1632	3321	3140
14	1215	3014	2404	2716	1632	3321	3140
15	793	2784	1215	793	1632	793	3140
16	395	2487	1215	793	1632	793	3140
17	48	2135	1215	793	1632	793	3140
18	0	1742	48	793	1632	793	0
19	0	1322	48	793	1632	793	0
20	0	897	48	0	0	793	0
21	0	491	0	0	0	793	0
22	0	129	0	0	0	793	0
23	0	0	0	0	0	793	0
24	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0

33	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0
37	0	0	0	0	0	0	0
38	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0
41	0	0	0	0	0	0	0
42	0	0	0	0	0	0	0
43	0	0	0	0	0	0	0
44	0	0	0	0	0	0	0
45	0	0	0	0	0	0	0
46	0	0	0	0	0	0	0
47	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0
49	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0
51	0	0	0	0	0	0	0
52	0	0	0	0	0	0	0
53	0	0	0	0	0	0	0
54	0	0	0	0	0	0	0
55	0	0	0	0	0	0	0
56	0	0	0	0	0	0	0
57	0	0	0	0	0	0	0
58	0	0	0	0	0	0	0
59	112	0	0	0	0	0	0
60	356	0	112	356	0	356	0
61	667	0	112	356	0	356	0
62	1036	0	112	356	1036	356	1855
63	1436	0	1036	356	1036	356	1855
64	1855	61	1036	356	1036	356	1855
65	2275	289	1036	2275	1036	356	1855
66	2671	583	2275	2275	1036	356	1855
67	3026	940	2275	2275	1036	356	1855
68	3321	1334	2275	2275	1036	356	1855
69	3552	1749	3321	2275	3552	3552	1855
70	3715	2172	3321	3715	3552	3552	1855
71	3812	2576	3321	3715	3552	3552	1855
72	3851	2942	3812	3715	3552	3552	1855
73	3838	3253	3812	3715	3552	3552	3705
74	3784	3501	3812	3715	3552	3552	3705
75	3705	3682	3784	3705	3552	3552	3705

76	3612	3794	3784	3705	3612	3552	3705
77	3515	3847	3784	3705	3612	3552	3705
78	3424	3846	3515	3705	3424	3424	3705
79	3345	3801	3515	3705	3424	3424	3705
80	3284	3727	3515	3284	3424	3424	3705
81	3243	3636	3284	3284	3424	3424	3705
82	3221	3539	3284	3284	3424	3424	3705
83	3219	3445	3284	3284	3424	3424	3705
84	3232	3363	3219	3284	3424	3424	3285
85	3254	3297	3219	3254	3254	3424	3285
86	3285	3251	3219	3254	3254	3424	3285
87	3319	3225	3285	3254	3254	3319	3285
88	3352	3218	3285	3254	3254	3319	3285
89	3381	3228	3285	3254	3254	3319	3285
90	3405	3248	3381	3405	3254	3319	3285
91	3423	3276	3381	3405	3254	3319	3285
92	3434	3310	3381	3405	3434	3319	3285
93	3438	3344	3434	3405	3434	3319	3285
94	3435	3374	3434	3405	3434	3319	3285
95	3429	3400	3434	3429	3434	3319	3407
96	3419	3419	3429	3429	3434	3419	3407
97	3407	3432	3429	3429	3434	3419	3407
98	3395	3437	3429	3429	3434	3419	3407
99	3383	3437	3395	3429	3383	3419	3407
100	3373	3432	3395	3373	3383	3419	3407
101	3367	3422	3395	3373	3383	3419	3407
102	3361	3410	3367	3373	3383	3419	3407
103	3359	3398	3367	3373	3383	3419	3407
104	3356	3386	3367	3373	3383	3419	3407
105	3349	3375	3356	3349	3383	3349	3407
106	3321	3368	3356	3349	3321	3349	3140
107	3256	3362	3356	3349	3321	3349	3140
108	3140	3359	3256	3349	3321	3349	3140
109	2962	3357	3256	3349	3321	3349	3140
110	2715	3351	3256	2715	3321	3349	3140
111	2404	3331	2715	2715	3321	3349	3140
112	2041	3276	2715	2715	3321	3349	3140
113	1640	3175	2715	2715	1640	3349	3140
114	1214	3013	1640	2715	1640	1214	3140
115	793	2783	1640	793	1640	1214	3140
116	395	2487	1640	793	1640	1214	3140
117	47	2136	395	793	1640	1214	0
118	0	1743	395	793	1640	1214	0

119	**0**	**1322**	**395**	**793**	**1640**	**1214**	**0**
120	**0**	**897**	**0**	**0**	**0**	**1214**	**0**
121	**0**	**491**	**0**	**0**	**0**	**1214**	**0**
122	**0**	**128**	**0**	**0**	**0**	**1214**	**0**
123	**0**	**0**	**0**	**0**	**0**	**0**	**0**
124	**0**	**0**	**0**	**0**	**0**	**0**	**0**
125	**0**	**0**	**0**	**0**	**0**	**0**	**0**
126	**0**	**0**	**0**	**0**	**0**	**0**	**0**
127	**0**	**0**	**0**	**0**	**0**	**0**	**0**
128	**0**	**0**	**0**	**0**	**0**	**0**	**0**
129	**0**	**0**	**0**	**0**	**0**	**0**	**0**
130	**0**	**0**	**0**	**0**	**0**	**0**	**0**
131	**0**	**0**	**0**	**0**	**0**	**0**	**0**
132	**0**	**0**	**0**	**0**	**0**	**0**	**0**
133	**0**	**0**	**0**	**0**	**0**	**0**	**0**
134	**0**	**0**	**0**	**0**	**0**	**0**	**0**
135	**0**	**0**	**0**	**0**	**0**	**0**	**0**
136	**0**	**0**	**0**	**0**	**0**	**0**	**0**
137	**0**	**0**	**0**	**0**	**0**	**0**	**0**
138	**2**	**0**	**0**	**0**	**0**	**0**	**0**
139	**0**	**0**	**0**	**0**	**0**	**0**	**0**
140	**0**	**0**	**0**	**0**	**0**	**0**	**0**
141	**0**	**0**	**0**	**0**	**0**	**0**	**0**
142	**1**	**0**	**0**	**0**	**0**	**0**	**0**
143	**1**	**2**	**0**	**0**	**0**	**0**	**0**
144	**0**	**0**	**1**	**0**	**0**	**0**	**0**
145	**3**	**0**	**1**	**3**	**0**	**0**	**0**
146	**0**	**0**	**1**	**3**	**0**	**0**	**0**
147	**0**	**0**	**0**	**3**	**0**	**0**	**0**
148	**0**	**1**	**0**	**3**	**0**	**0**	**0**
149	**0**	**0**	**0**	**3**	**0**	**0**	**0**
150	**0**	**2**	**0**	**0**	**0**	**0**	**0**
151	**0**	**1**	**0**	**0**	**0**	**0**	**0**
152	**0**	**0**	**0**	**0**	**0**	**0**	**0**
153	**0**	**0**	**0**	**0**	**0**	**0**	**0**
154	**0**	**0**	**0**	**0**	**0**	**0**	**0**
155	**0**	**0**	**0**	**0**	**0**	**0**	**0**
156	**0**	**0**	**0**	**0**	**0**	**0**	**0**
157	**0**	**0**	**0**	**0**	**0**	**0**	**0**
158	**0**	**0**	**0**	**0**	**0**	**0**	**0**
159	**0**	**0**	**0**	**0**	**0**	**0**	**0**
160	**0**	**0**	**0**	**0**	**0**	**0**	**0**
161	**0**	**0**	**0**	**0**	**0**	**0**	**0**

162	**0**	**0**	**0**	**0**	**0**	**0**	**0**
163	**0**	**0**	**0**	**0**	**0**	**0**	**0**
164	**0**	**0**	**0**	**0**	**0**	**0**	**0**
165	**0**	**0**	**0**	**0**	**0**	**0**	**0**
166	**0**	**0**	**0**	**0**	**0**	**0**	**0**
167	**0**	**0**	**0**	**0**	**0**	**0**	**0**
168	**0**	**0**	**0**	**0**	**0**	**0**	**0**
169	**0**	**0**	**0**	**0**	**0**	**0**	**0**
170	**0**	**0**	**0**	**0**	**0**	**0**	**0**
171	**0**	**0**	**0**	**0**	**0**	**0**	**0**
172	**0**	**0**	**0**	**0**	**0**	**0**	**0**
173	**0**	**0**	**0**	**0**	**0**	**0**	**0**
174	**0**	**0**	**0**	**0**	**0**	**0**	**0**
175	**0**	**0**	**0**	**0**	**0**	**0**	**0**
176	**0**	**0**	**0**	**0**	**0**	**0**	**0**
177	**0**	**0**	**0**	**0**	**0**	**0**	**0**
178	**0**	**0**	**0**	**0**	**0**	**0**	**0**
179	**0**	**0**	**0**	**0**	**0**	**0**	**0**
180	**0**	**0**	**0**	**0**	**0**	**0**	**0**
181	**0**	**0**	**0**	**0**	**0**	**0**	**0**
182	**0**	**0**	**0**	**0**	**0**	**0**	**0**
183	**0**	**0**	**0**	**0**	**0**	**0**	**0**
184	**0**	**0**	**0**	**0**	**0**	**0**	**0**
185	**0**	**0**	**0**	**0**	**0**	**0**	**0**
186	**0**	**0**	**0**	**0**	**0**	**0**	**0**
187	**0**	**0**	**0**	**0**	**0**	**0**	**0**
188	**0**	**0**	**0**	**0**	**0**	**0**	**0**
189	**0**	**0**	**0**	**0**	**0**	**0**	**0**
190	**0**	**0**	**0**	**0**	**0**	**0**	**0**
191	**0**	**0**	**0**	**0**	**0**	**0**	**0**
192	**0**	**0**	**0**	**0**	**0**	**0**	**0**
193	**0**	**0**	**0**	**0**	**0**	**0**	**0**
194	**0**	**0**	**0**	**0**	**0**	**0**	**0**
195	**0**	**0**	**0**	**0**	**0**	**0**	**0**
196	**0**	**0**	**0**	**0**	**0**	**0**	**0**
197	**99**	**0**	**0**	**0**	**99**	**0**	**0**
198	**351**	**0**	**99**	**0**	**99**	**0**	**0**
199	**672**	**0**	**99**	**0**	**99**	**0**	**0**
200	**1036**	**0**	**99**	**1036**	**99**	**0**	**0**
201	**1441**	**0**	**1036**	**1036**	**99**	**0**	**0**
202	**1864**	**50**	**1036**	**1036**	**99**	**0**	**0**
203	**2284**	**280**	**1036**	**1036**	**99**	**0**	**0**
204	**2680**	**587**	**2284**	**1036**	**2680**	**2680**	**0**

205	**3028**	**941**	**2284**	**3028**	**2680**	**2680**	**3555**
206	**3323**	**1337**	**2284**	**3028**	**2680**	**2680**	**3555**
207	**3555**	**1758**	**3323**	**3028**	**2680**	**2680**	**3555**
208	**3717**	**2181**	**3323**	**3028**	**2680**	**2680**	**3555**
209	**3813**	**2585**	**3323**	**3028**	**2680**	**2680**	**3555**
210	**3849**	**2946**	**3813**	**3849**	**2680**	**2680**	**3555**
211	**3835**	**3255**	**3813**	**3849**	**3835**	**2680**	**3555**
212	**3782**	**3504**	**3813**	**3849**	**3835**	**2680**	**3555**
213	**3704**	**3682**	**3782**	**3849**	**3835**	**3704**	**3555**
214	**3610**	**3795**	**3782**	**3849**	**3835**	**3704**	**3555**
215	**3512**	**3845**	**3782**	**3849**	**3835**	**3704**	**3555**
216	**3422**	**3842**	**3512**	**3849**	**3835**	**3704**	**3283**
217	**3344**	**3798**	**3512**	**3849**	**3835**	**3704**	**3283**
218	**3283**	**3725**	**3512**	**3849**	**3283**	**3704**	**3283**
219	**3242**	**3634**	**3283**	**3849**	**3283**	**3704**	**3283**
220	**3223**	**3536**	**3283**	**3223**	**3283**	**3704**	**3283**
221	**3220**	**3443**	**3283**	**3223**	**3283**	**3704**	**3283**
222	**3232**	**3362**	**3220**	**3223**	**3283**	**3232**	**3283**
223	**3255**	**3296**	**3220**	**3223**	**3283**	**3232**	**3283**
224	**3286**	**3250**	**3220**	**3223**	**3283**	**3232**	**3283**
225	**3319**	**3226**	**3286**	**3319**	**3319**	**3232**	**3283**
226	**3352**	**3219**	**3286**	**3319**	**3319**	**3232**	**3283**
227	**3382**	**3228**	**3286**	**3319**	**3319**	**3232**	**3424**
228	**3406**	**3248**	**3382**	**3319**	**3319**	**3232**	**3424**
229	**3424**	**3278**	**3382**	**3319**	**3319**	**3232**	**3424**
230	**3434**	**3311**	**3382**	**3434**	**3319**	**3232**	**3424**
231	**3437**	**3343**	**3434**	**3434**	**3319**	**3437**	**3424**
232	**3435**	**3375**	**3434**	**3434**	**3435**	**3437**	**3424**
233	**3428**	**3400**	**3434**	**3434**	**3435**	**3437**	**3424**
234	**3418**	**3420**	**3428**	**3434**	**3435**	**3437**	**3424**
235	**3407**	**3432**	**3428**	**3407**	**3435**	**3437**	**3424**
236	**3395**	**3437**	**3428**	**3407**	**3435**	**3437**	**3424**
237	**3384**	**3437**	**3395**	**3407**	**3435**	**3437**	**3424**
238	**3374**	**3430**	**3395**	**3407**	**3435**	**3437**	**3361**
239	**3366**	**3421**	**3395**	**3407**	**3366**	**3437**	**3361**
240	**3361**	**3410**	**3366**	**3361**	**3366**	**3361**	**3361**
241	**3358**	**3398**	**3366**	**3361**	**3366**	**3361**	**3361**
242	**3355**	**3387**	**3366**	**3361**	**3366**	**3361**	**3361**
243	**3349**	**3376**	**3355**	**3361**	**3366**	**3361**	**3361**
244	**3321**	**3367**	**3355**	**3361**	**3366**	**3361**	**3361**
245	**3255**	**3362**	**3355**	**3255**	**3366**	**3361**	**3361**
246	**3142**	**3359**	**3255**	**3255**	**3142**	**3361**	**3361**
247	**2963**	**3356**	**3255**	**3255**	**3142**	**3361**	**3361**

248	2714	3352	3255	3255	3142	3361	3361
249	2404	3331	2714	3255	3142	2404	1638
250	2041	3275	2714	2041	3142	2404	1638
251	1638	3176	2714	2041	3142	2404	1638
252	1214	3015	1638	2041	3142	2404	1638
253	793	2782	1638	2041	793	2404	1638
254	395	2486	1638	2041	793	2404	1638

Table 33. *Quantization error noise generated by 11th odd order sampling components*

Noise(N+1)3	Noise(N+1)5	Noise(N+1)7	Noise(N+1)9	Noise(N+1)11
-10	0	-10	-32	-32
-18	-8	-18	-40	-40
-22	-12	-22	-44	-44
-2	-14	-24	-46	-46
-6	-18	-28	-50	-50
-14	0	-36	-58	-58
-27	-27	0	0	181
-92	-92	-65	-65	116
-208	-208	-181	-181	0
-177	-385	-358	-358	-177
-424	0	-605	-605	-424
-736	-312	-917	-917	-736
-364	-676	-1281	-1281	-1100
-766	-1078	6	-1683	-1502
-1189	-1501	-417	-2106	-1925
-422	0	-839	0	-2347
-820	-398	-1237	-398	-2745
-1167	-745	-1584	-745	-3092
-48	-793	-1632	-793	0
-48	-793	-1632	-793	0
-48	0	0	-793	0
0	0	0	-793	0
0	0	0	-793	0
0	0	0	-793	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**
112	**112**	**112**	**112**	**112**
244	**0**	**356**	**0**	**356**
555	**311**	**667**	**311**	**667**
924	**680**	**0**	**680**	**-819**
400	**1080**	**400**	**1080**	**-419**
819	**1499**	**819**	**1499**	**0**
1239	**0**	**1239**	**1919**	**420**
396	**396**	**1635**	**2315**	**816**
751	**751**	**1990**	**2670**	**1171**
1046	**1046**	**2285**	**2965**	**1466**
231	**1277**	**0**	**0**	**1697**
394	**0**	**163**	**163**	**1860**
491	**97**	**260**	**260**	**1957**
39	**136**	**299**	**299**	**1996**
26	**123**	**286**	**286**	**133**
-28	**69**	**232**	**232**	**79**
-79	**0**	**153**	**153**	**0**

-172	-93	0	60	-93
-269	-190	-97	-37	-190
-91	-281	0	0	-281
-170	-360	-79	-79	-360
-231	0	-140	-140	-421
-41	-41	-181	-181	-462
-63	-63	-203	-203	-484
-65	-65	-205	-205	-486
13	-52	-192	-192	-53
35	0	0	-170	-31
66	31	31	-139	0
34	65	65	0	34
67	98	98	33	67
96	127	127	62	96
24	0	151	86	120
42	18	169	104	138
53	29	0	115	149
4	33	4	119	153
1	30	1	116	150
-5	0	-5	110	22
-10	-10	-15	0	12
-22	-22	-27	-12	0
-34	-34	-39	-24	-12
-12	-46	0	-36	-24
-22	0	-10	-46	-34
-28	-6	-16	-52	-40
-6	-12	-22	-58	-46
-8	-14	-24	-60	-48
-11	-17	-27	-63	-51
-7	0	-34	0	-58
-35	-28	0	-28	181
-100	-93	-65	-93	116
-116	-209	-181	-209	0
-294	-387	-359	-387	-178
-541	0	-606	-634	-425
-311	-311	-917	-945	-736
-674	-674	-1280	-1308	-1099
-1075	-1075	0	-1709	-1500
-426	-1501	-426	0	-1926
-847	0	-847	-421	-2347
-1245	-398	-1245	-819	-2745
-348	-746	-1593	-1167	47
-395	-793	-1640	-1214	0
-395	-793	-1640	-1214	0

0	0	0	-1214	0
0	0	0	-1214	0
0	0	0	-1214	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
2	2	2	2	2
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
1	1	1	1	1
1	1	1	1	1
-1	0	0	0	0
2	0	3	3	3
-1	-3	0	0	0
0	-3	0	0	0
0	-3	0	0	0
0	-3	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
99	99	0	99	99
252	351	252	351	351
573	672	573	672	672
937	0	937	1036	1036
405	405	1342	1441	1441
828	828	1765	1864	1864
1248	1248	2185	2284	2284
396	1644	0	0	2680
744	0	348	348	-527
1039	295	643	643	-232
232	527	875	875	0

394	**689**	**1037**	**1037**	**162**
490	**785**	**1133**	**1133**	**258**
36	**0**	**1169**	**1169**	**294**
22	**-14**	**0**	**1155**	**280**
-31	**-67**	**-53**	**1102**	**227**
-78	**-145**	**-131**	**0**	**149**
-172	**-239**	**-225**	**-94**	**55**
-270	**-337**	**-323**	**-192**	**-43**
-90	**-427**	**-413**	**-282**	**139**
-168	**-505**	**-491**	**-360**	**61**
-229	**-566**	**0**	**-421**	**0**
-41	**-607**	**-41**	**-462**	**-41**
-60	**0**	**-60**	**-481**	**-60**
-63	**-3**	**-63**	**-484**	**-63**
12	**9**	**-51**	**0**	**-51**
35	**32**	**-28**	**23**	**-28**
66	**63**	**3**	**54**	**3**
33	**0**	**0**	**87**	**36**
66	**33**	**33**	**120**	**69**
96	**63**	**63**	**150**	**-42**
24	**87**	**87**	**174**	**-18**
42	**105**	**105**	**192**	**0**
52	**0**	**115**	**202**	**10**
3	**3**	**118**	**0**	**13**
1	**1**	**0**	**-2**	**11**
-6	**-6**	**-7**	**-9**	**4**
-10	**-16**	**-17**	**-19**	**-6**
-21	**0**	**-28**	**-30**	**-17**
-33	**-12**	**-40**	**-42**	**-29**
-11	**-23**	**-51**	**-53**	**-40**
-21	**-33**	**-61**	**-63**	**13**
-29	**-41**	**0**	**-71**	**5**
-5	**0**	**-5**	**0**	**0**
-8	**-3**	**-8**	**-3**	**-3**
-11	**-6**	**-11**	**-6**	**-6**
-6	**-12**	**-17**	**-12**	**-12**
-34	**-40**	**-45**	**-40**	**-40**
-100	**0**	**-111**	**-106**	**-106**
-113	**-113**	**0**	**-219**	**-219**
-292	**-292**	**-179**	**-398**	**-398**
-541	**-541**	**-428**	**-647**	**-647**
-310	**-851**	**-738**	**0**	**766**
-673	**0**	**-1101**	**-363**	**403**
-1076	**-403**	**-1504**	**-766**	**0**

-424	**-827**	**-1928**	**-1190**	**-424**
-845	**-1248**	**0**	**-1611**	**-845**
-1243	**-1646**	**-398**	**-2009**	**-1243**

Table 34. *Power calculation between fundamental and odd order harmonics*

S	N(N+1)3	N(N+1)5	N(N+1)7	N(N+1)9	N(N+1)11
11383876	100	0	100	1024	1024
11329956	324	64	324	1600	1600
11303044	484	144	484	1936	1936
11289600	4	196	576	2116	2116
11262736	36	324	784	2500	2500
11209104	196	0	1296	3364	3364
11029041	729	729	0	0	32761
10601536	8464	8464	4225	4225	13456
9859600	43264	43264	32761	32761	0
8779369	31329	148225	128164	128164	31329
7376656	179776	0	366025	366025	179776
5779216	541696	97344	840889	840889	541696
4161600	132496	456976	1640961	1640961	1210000
2683044	586756	1162084	36	2832489	2256004
1476225	1413721	2253001	173889	4435236	3705625
628849	178084	0	703921	0	5508409
156025	672400	158404	1530169	158404	7535025
2304	1361889	555025	2509056	555025	9560464
0	2304	628849	2663424	628849	0
0	2304	628849	2663424	628849	0
0	2304	0	0	628849	0
0	0	0	0	628849	0
0	0	0	0	628849	0
0	0	0	0	628849	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
12544	**12544**	**12544**	**12544**	**12544**	**12544**
126736	**59536**	**0**	**126736**	**0**	**126736**
444889	**308025**	**96721**	**444889**	**96721**	**444889**
1073296	**853776**	**462400**	**0**	**462400**	**670761**
2062096	**160000**	**1166400**	**160000**	**1166400**	**175561**
3441025	**670761**	**2247001**	**670761**	**2247001**	**0**
5175625	**1535121**	**0**	**1535121**	**3682561**	**176400**
7134241	**156816**	**156816**	**2673225**	**5359225**	**665856**
9156676	**564001**	**564001**	**3960100**	**7128900**	**1371241**
11029041	**1094116**	**1094116**	**5221225**	**8791225**	**2149156**
12616704	**53361**	**1630729**	**0**	**0**	**2879809**
13801225	**155236**	**0**	**26569**	**26569**	**3459600**
14531344	**241081**	**9409**	**67600**	**67600**	**3829849**
14830201	**1521**	**18496**	**89401**	**89401**	**3984016**
14730244	**676**	**15129**	**81796**	**81796**	**17689**
14318656	**784**	**4761**	**53824**	**53824**	**6241**
13727025	**6241**	**0**	**23409**	**23409**	**0**
13046544	**29584**	**8649**	**0**	**3600**	**8649**
12355225	**72361**	**36100**	**9409**	**1369**	**36100**
11723776	**8281**	**78961**	**0**	**0**	**78961**
11189025	**28900**	**129600**	**6241**	**6241**	**129600**
10784656	**53361**	**0**	**19600**	**19600**	**177241**
10517049	**1681**	**1681**	**32761**	**32761**	**213444**
10374841	**3969**	**3969**	**41209**	**41209**	**234256**
10361961	**4225**	**4225**	**42025**	**42025**	**236196**
10445824	**169**	**2704**	**36864**	**36864**	**2809**
10588516	**1225**	**0**	**0**	**28900**	**961**

10791225	4356	961	961	19321	0
11015761	1156	4225	4225	0	1156
11235904	4489	9604	9604	1089	4489
11431161	9216	16129	16129	3844	9216
11594025	576	0	22801	7396	14400
11716929	1764	324	28561	10816	19044
11792356	2809	841	0	13225	22201
11819844	16	1089	16	14161	23409
11799225	1	900	1	13456	22500
11758041	25	0	25	12100	484
11689561	100	100	225	0	144
11607649	484	484	729	144	0
11526025	1156	1156	1521	576	144
11444689	144	2116	0	1296	576
11377129	484	0	100	2116	1156
11336689	784	36	256	2704	1600
11296321	36	144	484	3364	2116
11282881	64	196	576	3600	2304
11262736	121	289	729	3969	2601
11215801	49	0	1156	0	3364
11029041	1225	784	0	784	32761
10601536	10000	8649	4225	8649	13456
9859600	13456	43681	32761	43681	0
8773444	86436	149769	128881	149769	31684
7371225	292681	0	367236	401956	180625
5779216	96721	96721	840889	893025	541696
4165681	454276	454276	1638400	1710864	1207801
2689600	1155625	1155625	0	2920681	2250000
1473796	181476	2253001	181476	0	3709476
628849	717409	0	717409	177241	5508409
156025	1550025	158404	1550025	670761	7535025
2209	121104	556516	2537649	1361889	2209
0	156025	628849	2689600	1473796	0
0	156025	628849	2689600	1473796	0
0	0	0	0	1473796	0
0	0	0	0	1473796	0
0	0	0	0	1473796	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
4	**4**	**4**	**4**	**4**	**4**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
1	**1**	**1**	**1**	**1**	**1**
1	**1**	**1**	**1**	**1**	**1**
0	**1**	**0**	**0**	**0**	**0**
9	**4**	**0**	**9**	**9**	**9**
0	**1**	**9**	**0**	**0**	**0**
0	**0**	**9**	**0**	**0**	**0**
0	**0**	**9**	**0**	**0**	**0**
0	**0**	**9**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
9801	9801	9801	0	9801	9801
123201	63504	123201	63504	123201	123201
451584	328329	451584	328329	451584	451584
1073296	877969	0	877969	1073296	1073296
2076481	164025	164025	1800964	2076481	2076481
3474496	685584	685584	3115225	3474496	3474496
5216656	1557504	1557504	4774225	5216656	5216656
7182400	156816	2702736	0	0	7182400
9168784	553536	0	121104	121104	277729
11042329	1079521	87025	413449	413449	53824
12638025	53824	277729	765625	765625	0
13816089	155236	474721	1075369	1075369	26244
14538969	240100	616225	1283689	1283689	66564
14814801	1296	0	1366561	1366561	86436
14707225	484	196	0	1334025	78400
14303524	961	4489	2809	1214404	51529
13719616	6084	21025	17161	0	22201
13032100	29584	57121	50625	8836	3025
12334144	72900	113569	104329	36864	1849
11710084	8100	182329	170569	79524	19321
11182336	28224	255025	241081	129600	3721

10778089	**52441**	**320356**	**0**	**177241**	**0**
10510564	**1681**	**368449**	**1681**	**213444**	**1681**
10387729	**3600**	**0**	**3600**	**231361**	**3600**
10368400	**3969**	**9**	**3969**	**234256**	**3969**
10445824	**144**	**81**	**2601**	**0**	**2601**
10595025	**1225**	**1024**	**784**	**529**	**784**
10797796	**4356**	**3969**	**9**	**2916**	**9**
11015761	**1089**	**0**	**0**	**7569**	**1296**
11235904	**4356**	**1089**	**1089**	**14400**	**4761**
11437924	**9216**	**3969**	**3969**	**22500**	**1764**
11600836	**576**	**7569**	**7569**	**30276**	**324**
11723776	**1764**	**11025**	**11025**	**36864**	**0**
11792356	**2704**	**0**	**13225**	**40804**	**100**
11812969	**9**	**9**	**13924**	**0**	**169**
11799225	**1**	**1**	**0**	**4**	**121**
11751184	**36**	**36**	**49**	**81**	**16**
11682724	**100**	**256**	**289**	**361**	**36**
11607649	**441**	**0**	**784**	**900**	**289**
11526025	**1089**	**144**	**1600**	**1764**	**841**
11451456	**121**	**529**	**2601**	**2809**	**1600**
11383876	**441**	**1089**	**3721**	**3969**	**169**
11329956	**841**	**1681**	**0**	**5041**	**25**
11296321	**25**	**0**	**25**	**0**	**0**
11276164	**64**	**9**	**64**	**9**	**9**
11256025	**121**	**36**	**121**	**36**	**36**
11215801	**36**	**144**	**289**	**144**	**144**
11029041	**1156**	**1600**	**2025**	**1600**	**1600**
10595025	**10000**	**0**	**12321**	**11236**	**11236**
9872164	**12769**	**12769**	**0**	**47961**	**47961**
8779369	**85264**	**85264**	**32041**	**158404**	**158404**
7365796	**292681**	**292681**	**183184**	**418609**	**418609**
5779216	**96100**	**724201**	**544644**	**0**	**586756**
4165681	**452929**	**0**	**1212201**	**131769**	**162409**
2683044	**1157776**	**162409**	**2262016**	**586756**	**0**
1473796	**179776**	**683929**	**3717184**	**1416100**	**179776**
628849	**714025**	**1557504**	**0**	**2595321**	**714025**
156025	**1545049**	**2709316**	**158404**	**4036081**	**1545049**
1212658261	**26967691**	**34795150**	**66831677**	**90789110**	**96968638**

Table 35. *RMS calculation for fundamental and odd order harmonics*

Srms	Nrms(N+1)3	Nrms(N+1)5	Nrms(N+1)7	Nrms(N+1)9	Nrms(N+1)11

	136,56	**20,36**	**23,13**	**32,06**	**37,37**	**38,62**

Table 36. *SNR calculations for all the sampling schemes investigated in this dissertation*

BIT's	odd/even	Noise Jamming	Gaussian Jamming	Mod-Delta	Mod-Delta(Gaussian)	Mod-Delta(Noise)
2	**10,03**	**9,17**	**13,31**	**11,89**	**10,22**	**6,47**
4	**8,92**	**11,31**	**14,70**	**0,92**	**3,86**	**11,21**
6	**6,09**	**12,01**	**11,16**	**2,03**	**6,79**	**12,15**
8	**4,76**	**11,15**	**14,71**	**4,50**	**5,02**	**7,02**
10	**4,47**	**12,23**	**13,32**	**9,21**	**0,41**	**4,92**
12	**4,32**	**10,14**	**12,24**	**17,80**	**5,96**	**5,80**

Appendix B: Mod-Δ Data Analysis

#	I	Q	MD(N+1)3		MD(N+1)5	MD(N+1)7	MD(N+1)9	MD(N+1)11	MD(N+1)13
0	3374	3432	66	44	0	2	0	56	78
1	3366	3422	56	37	4	8	28	46	66
2	3362	3410	46	31	4	14	20	38	56
3	3360	3397	38	25	1	6	25	34	46
4	3356	3386	34	23	0	2	24	32	38
5	3348	3376	32	21	4	0	4	28	34
6	3321	3367	28	19	3	12	19	20	32
7	3256	3362	20	13	4	4	28	57	28
8	3140	3360	121	81	4	9	4	56	20
9	2963	3357	56	37	0	8	24	4	121
10	2716	3351	68	45	6	4	6	19	56
11	2404	3331	19	13	7	3	31	28	68
12	2040	3276	28	19	1	12	25	36	19
13	1638	3175	100	67	3	4	11	56	28
14	1215	3014	120	80	0	8	16	38	100
15	793	2784	102	68	0	6	0	63	120
16	395	2487	63	42	0	15	0	25	102
17	48	2135	25	17	0	9	0	11	63
18	0	1742	11	7	0	11	0	48	25
19	0	1322	48	32	0	0	0	0	11
20	0	897	0	0	0	0	0	0	48
21	0	491	0	0	0	0	0	0	0
22	0	129	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0
37	0	0	0	0	0	0	0	0	0

38	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
39	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
40	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
41	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
42	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
43	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
44	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
45	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
46	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
47	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
48	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
49	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
50	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
51	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
52	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
53	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
54	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
55	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
56	**0**	**0**	**0**	**0**	**0**	**0**	**16**	**0**	**0**
57	**0**	**0**	**0**	**0**	**4**	**0**	**4**	**0**	**0**
58	**0**	**0**	**0**	**0**	**3**	**0**	**27**	**0**	**0**
59	**112**	**0**	**0**	**0**	**4**	**0**	**12**	**0**	**0**
60	**356**	**0**	**0**	**0**	**4**	**0**	**28**	**48**	**0**
61	**667**	**0**	**112**	**75**	**7**	**0**	**31**	**36**	**0**
62	**1036**	**0**	**100**	**67**	**3**	**4**	**3**	**27**	**112**
63	**1436**	**0**	**27**	**18**	**7**	**11**	**15**	**12**	**100**
64	**1855**	**61**	**12**	**8**	**2**	**12**	**18**	**28**	**27**
65	**2275**	**289**	**28**	**19**	**1**	**12**	**25**	**63**	**12**
66	**2671**	**583**	**63**	**42**	**0**	**15**	**0**	**35**	**28**
67	**3026**	**940**	**99**	**66**	**3**	**3**	**3**	**47**	**63**
68	**3321**	**1334**	**111**	**74**	**4**	**15**	**4**	**18**	**99**
69	**3552**	**1749**	**82**	**55**	**3**	**2**	**11**	**57**	**111**
70	**3715**	**2172**	**121**	**81**	**6**	**9**	**30**	**32**	**82**
71	**3812**	**2576**	**96**	**64**	**0**	**0**	**8**	**3**	**121**
72	**3851**	**2942**	**3**	**2**	**1**	**3**	**25**	**36**	**96**
73	**3838**	**3253**	**100**	**67**	**4**	**4**	**28**	**11**	**3**
74	**3784**	**3501**	**11**	**7**	**3**	**11**	**27**	**62**	**100**
75	**3705**	**3682**	**126**	**84**	**0**	**14**	**0**	**8**	**11**
76	**3612**	**3794**	**72**	**48**	**1**	**8**	**17**	**57**	**126**
77	**3515**	**3847**	**121**	**81**	**4**	**9**	**20**	**28**	**72**
78	**3424**	**3846**	**28**	**19**	**3**	**12**	**11**	**59**	**121**
79	**3345**	**3801**	**59**	**39**	**5**	**11**	**21**	**32**	**28**
80	**3284**	**3727**	**96**	**64**	**3**	**0**	**19**	**17**	**59**

81	3243	3636	17	11	0	1	0	20	96
82	3221	3539	84	56	6	4	22	43	17
83	3219	3445	43	29	5	11	21	21	84
84	3232	3363	21	14	7	5	23	19	43
85	3254	3297	19	13	0	3	24	32	21
86	3285	3251	32	21	5	0	21	54	19
87	3319	3225	54	36	5	6	13	21	32
88	3352	3218	85	57	7	5	31	55	54
89	3381	3228	119	79	2	7	10	24	85
90	3405	3248	24	16	6	8	14	53	119
91	3423	3276	53	35	3	5	11	13	24
92	3434	3310	77	51	5	13	5	31	53
93	3438	3344	95	63	3	15	27	42	77
94	3435	3374	106	71	7	10	15	46	95
95	3429	3400	110	73	3	14	3	43	106
96	3419	3419	107	71	7	11	23	37	110
97	3407	3432	101	67	5	5	13	27	107
98	3395	3437	91	61	7	11	7	15	101
99	3383	3437	79	53	1	15	1	3	91
100	3373	3432	67	45	7	3	31	55	79
101	3367	3422	55	37	4	7	28	45	67
102	3361	3410	45	30	5	13	21	39	55
103	3359	3398	39	26	1	7	25	33	45
104	3356	3386	33	22	0	1	24	31	39
105	3349	3375	31	21	4	15	4	28	33
106	3321	3368	28	19	2	12	18	21	31
107	3256	3362	21	14	3	5	27	57	28
108	3140	3359	121	81	4	9	4	56	21
109	2962	3357	56	37	1	8	25	4	121
110	2715	3351	68	45	0	4	8	18	56
111	2404	3331	18	12	6	2	30	27	68
112	2041	3276	27	18	1	11	25	36	18
113	1640	3175	100	67	3	4	11	57	27
114	1214	3013	121	81	7	9	15	40	100
115	793	2783	104	69	0	8	0	62	121
116	395	2487	62	41	0	14	0	25	104
117	47	2136	25	17	0	9	0	11	62
118	0	1743	11	7	0	11	0	47	25
119	0	1322	47	31	0	15	0	0	11
120	0	897	0	0	0	0	0	0	47
121	0	491	0	0	0	0	0	0	0
122	0	128	0	0	0	0	0	0	0
123	0	0	0	0	0	0	0	0	0

124	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
125	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
126	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
127	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
128	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
129	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
130	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
131	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
132	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
133	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
134	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
135	**0**	**0**	**0**	**0**	**2**	**0**	**2**	**0**	**0**
136	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
137	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
138	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
139	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**2**	**0**
140	**0**	**0**	**2**	**1**	**1**	**2**	**1**	**0**	**0**
141	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**2**
142	**1**	**0**	**0**	**0**	**3**	**0**	**3**	**0**	**0**
143	**1**	**2**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
144	**0**	**0**	**1**	**1**	**0**	**1**	**0**	**1**	**0**
145	**3**	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**
146	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**3**	**1**
147	**0**	**0**	**3**	**2**	**0**	**3**	**0**	**0**	**0**
148	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**3**
149	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
150	**0**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
151	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
152	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
153	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
154	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
155	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
156	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
157	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
158	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
159	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
160	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
161	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
162	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
163	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
164	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
165	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
166	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

167	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
168	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
169	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
170	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
171	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
172	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
173	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
174	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
175	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
176	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
177	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
178	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
179	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
180	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
181	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
182	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
183	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
184	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
185	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
186	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
187	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
188	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
189	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
190	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
191	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
192	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
193	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
194	**0**	**0**	**0**	**0**	**3**	**0**	**3**	**0**	**0**
195	**0**	**0**	**0**	**0**	**7**	**0**	**31**	**0**	**0**
196	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
197	**99**	**0**	**0**	**0**	**4**	**0**	**12**	**0**	**0**
198	**351**	**0**	**0**	**0**	**1**	**0**	**1**	**35**	**0**
199	**672**	**0**	**99**	**66**	**0**	**3**	**8**	**31**	**0**
200	**1036**	**0**	**95**	**63**	**4**	**15**	**12**	**32**	**99**
201	**1441**	**0**	**32**	**21**	**0**	**0**	**24**	**12**	**95**
202	**1864**	**50**	**12**	**8**	**4**	**12**	**20**	**33**	**32**
203	**2284**	**280**	**33**	**22**	**3**	**1**	**27**	**8**	**12**
204	**2680**	**587**	**72**	**48**	**3**	**8**	**3**	**44**	**33**
205	**3028**	**941**	**108**	**72**	**5**	**12**	**5**	**56**	**72**
206	**3323**	**1337**	**120**	**80**	**5**	**8**	**5**	**20**	**108**
207	**3555**	**1758**	**84**	**56**	**1**	**4**	**9**	**59**	**120**
208	**3717**	**2181**	**123**	**82**	**3**	**11**	**27**	**35**	**84**
209	**3813**	**2585**	**99**	**66**	**6**	**3**	**6**	**5**	**123**

210	3849	2946	5	3	0	5	24	37	99
211	3835	3255	101	67	2	5	26	9	5
212	3782	3504	9	6	0	9	24	59	101
213	3704	3682	123	82	6	11	30	6	9
214	3610	3795	70	47	0	6	16	56	123
215	3512	3845	120	80	3	8	19	26	70
216	3422	3842	26	17	2	10	10	56	120
217	3344	3798	56	37	7	8	23	30	26
218	3283	3725	94	63	4	14	20	16	56
219	3242	3634	16	11	0	0	0	19	94
220	3223	3536	83	55	7	3	23	42	16
221	3220	3443	42	28	6	10	22	23	83
222	3232	3362	23	15	7	7	23	20	42
223	3255	3296	20	13	0	4	24	32	23
224	3286	3250	32	21	6	0	22	55	20
225	3319	3226	55	37	6	7	14	22	32
226	3352	3219	86	57	0	6	0	55	55
227	3382	3228	119	79	2	7	10	24	86
228	3406	3248	24	16	5	8	13	54	119
229	3424	3278	54	36	3	6	11	14	24
230	3434	3311	78	52	4	14	4	32	54
231	3437	3343	96	64	2	0	26	42	78
232	3435	3375	106	71	7	10	15	45	96
233	3428	3400	109	73	3	13	3	43	106
234	3418	3420	107	71	0	11	24	36	109
235	3407	3432	100	67	6	4	14	26	107
236	3395	3437	90	60	6	10	6	15	100
237	3384	3437	79	53	1	15	1	3	90
238	3374	3430	67	45	6	3	30	56	79
239	3366	3421	56	37	3	8	27	46	67
240	3361	3410	46	31	5	14	21	38	56
241	3358	3398	38	25	1	6	25	33	46
242	3355	3387	33	22	7	1	23	30	38
243	3349	3376	30	20	6	14	6	27	33
244	3321	3367	27	18	3	11	19	21	30
245	3255	3362	21	14	2	5	26	57	27
246	3142	3359	121	81	4	9	4	55	21
247	2963	3356	55	37	1	7	25	6	121
248	2714	3352	70	47	6	6	6	19	55
249	2404	3331	19	13	6	3	30	26	70
250	2041	3275	26	17	1	10	25	36	19
251	1638	3176	100	67	3	4	11	57	26
252	1214	3015	121	81	7	9	15	38	100

253	793	2782	102	68	0	6	0	62	121
254	395	2486	62	41	0	14	0	25	102

Noise(N+1)3	Noise(N+1)5	Noise(N+1)7	Noise(N+1)9	Noise(N+1)11	Noise(N+1)13
3330	3374	3372	3374	3318	3296
3329	3362	3358	3338	3320	3300
3331	3358	3348	3342	3324	3306
3335	3359	3354	3335	3326	3314
3333	3356	3354	3332	3324	3318
3327	3344	3348	3344	3320	3314
3302	3318	3309	3302	3301	3289
3243	3252	3252	3228	3199	3228
3059	3136	3131	3136	3084	3120
2926	2963	2955	2939	2959	2842
2671	2710	2712	2710	2697	2660
2391	2397	2401	2373	2376	2336
2021	2039	2028	2015	2004	2021
1571	1635	1634	1627	1582	1610
1135	1215	1207	1199	1177	1115
725	793	787	793	730	673
353	395	380	395	370	293
31	48	39	48	37	-15
-7	0	-11	0	-48	-25
-32	0	0	0	0	-11
0	0	0	0	0	-48
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	-16	0	0
0	-4	0	-4	0	0
0	-3	0	-27	0	0
112	108	112	100	112	112
356	352	356	328	308	356
592	660	667	636	631	667
969	1033	1032	1033	1009	924
1418	1429	1425	1421	1424	1336
1847	1853	1843	1837	1827	1828
2256	2274	2263	2250	2212	2263
2629	2671	2656	2671	2636	2643
2960	3023	3023	3023	2979	2963
3247	3317	3306	3317	3303	3222
3497	3549	3550	3541	3495	3441
3634	3709	3706	3685	3683	3633
3748	3812	3812	3804	3809	3691
3849	3850	3848	3826	3815	3755
3771	3834	3834	3810	3827	3835
3777	3781	3773	3757	3722	3684
3621	3705	3691	3705	3697	3694
3564	3611	3604	3595	3555	3486
3434	3511	3506	3495	3487	3443
3405	3421	3412	3413	3365	3303
3306	3340	3334	3324	3313	3317
3220	3281	3284	3265	3267	3225
3232	3243	3242	3243	3223	3147

3165	**3215**	**3217**	**3199**	**3178**	**3204**
3190	**3214**	**3208**	**3198**	**3198**	**3135**
3218	**3225**	**3227**	**3209**	**3213**	**3189**
3241	**3254**	**3251**	**3230**	**3222**	**3233**
3264	**3280**	**3285**	**3264**	**3231**	**3266**
3283	**3314**	**3313**	**3306**	**3298**	**3287**
3295	**3345**	**3347**	**3321**	**3297**	**3298**
3302	**3379**	**3374**	**3371**	**3357**	**3296**
3389	**3399**	**3397**	**3391**	**3352**	**3286**
3388	**3420**	**3418**	**3412**	**3410**	**3399**
3383	**3429**	**3421**	**3429**	**3403**	**3381**
3375	**3435**	**3423**	**3411**	**3396**	**3361**
3364	**3428**	**3425**	**3420**	**3389**	**3340**
3356	**3426**	**3415**	**3426**	**3386**	**3323**
3348	**3412**	**3408**	**3396**	**3382**	**3309**
3340	**3402**	**3402**	**3394**	**3380**	**3300**
3334	**3388**	**3384**	**3388**	**3380**	**3294**
3330	**3382**	**3368**	**3382**	**3380**	**3292**
3328	**3366**	**3370**	**3342**	**3318**	**3294**
3330	**3363**	**3360**	**3339**	**3322**	**3300**
3331	**3356**	**3348**	**3340**	**3322**	**3306**
3333	**3358**	**3352**	**3334**	**3326**	**3314**
3334	**3356**	**3355**	**3332**	**3325**	**3317**
3328	**3345**	**3334**	**3345**	**3321**	**3316**
3302	**3319**	**3309**	**3303**	**3300**	**3290**
3242	**3253**	**3251**	**3229**	**3199**	**3228**
3059	**3136**	**3131**	**3136**	**3084**	**3119**
2925	**2961**	**2954**	**2937**	**2958**	**2841**
2670	**2715**	**2711**	**2707**	**2697**	**2659**
2392	**2398**	**2402**	**2374**	**2377**	**2336**
2023	**2040**	**2030**	**2016**	**2005**	**2023**
1573	**1637**	**1636**	**1629**	**1583**	**1613**
1133	**1207**	**1205**	**1199**	**1174**	**1114**
724	**793**	**785**	**793**	**731**	**672**
354	**395**	**381**	**395**	**370**	**291**
30	**47**	**38**	**47**	**36**	**-15**
-7	**0**	**-11**	**0**	**-47**	**-25**
-31	**0**	**-15**	**0**	**0**	**-11**
0	**0**	**0**	**0**	**0**	**-47**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**

0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**-2**	**0**	**-2**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
2	**2**	**2**	**2**	**2**	**2**
0	**-1**	**0**	**-1**	**-2**	**0**
-1	**-1**	**-2**	**-1**	**0**	**0**
0	**0**	**0**	**0**	**0**	**-2**
1	**-2**	**1**	**-2**	**1**	**1**
1	**1**	**1**	**1**	**0**	**1**
-1	**0**	**-1**	**0**	**-1**	**0**
2	**3**	**2**	**3**	**3**	**2**
0	**0**	**0**	**0**	**-3**	**-1**
-2	**0**	**-3**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**-3**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	-3	0	-3	0	0
0	-7	0	-31	0	0
0	0	0	0	0	0
99	95	99	87	99	99
351	350	351	350	316	351
606	672	669	664	641	672
973	1032	1021	1024	1004	937
1420	1441	1441	1417	1429	1346
1856	1860	1852	1844	1831	1832
2262	2281	2283	2257	2276	2272
2632	2677	2672	2677	2636	2647
2956	3023	3016	3023	2972	2956
3243	3318	3315	3318	3303	3215
3499	3554	3551	3546	3496	3435
3635	3714	3706	3690	3682	3633
3747	3807	3810	3807	3808	3690
3846	3849	3844	3825	3812	3750
3768	3833	3830	3809	3826	3830
3776	3782	3773	3758	3723	3681
3622	3698	3693	3674	3698	3695

3563	**3610**	**3604**	**3594**	**3554**	**3487**
3432	**3509**	**3504**	**3493**	**3486**	**3442**
3405	**3420**	**3412**	**3412**	**3366**	**3302**
3307	**3337**	**3336**	**3321**	**3314**	**3318**
3220	**3279**	**3269**	**3263**	**3267**	**3227**
3231	**3242**	**3242**	**3242**	**3223**	**3148**
3168	**3216**	**3220**	**3200**	**3181**	**3207**
3192	**3214**	**3210**	**3198**	**3197**	**3137**
3217	**3225**	**3225**	**3209**	**3212**	**3190**
3242	**3255**	**3251**	**3231**	**3223**	**3232**
3265	**3280**	**3286**	**3264**	**3231**	**3266**
3282	**3313**	**3312**	**3305**	**3297**	**3287**
3295	**3352**	**3346**	**3352**	**3297**	**3297**
3303	**3380**	**3375**	**3372**	**3358**	**3296**
3390	**3401**	**3398**	**3393**	**3352**	**3287**
3388	**3421**	**3418**	**3413**	**3410**	**3400**
3382	**3430**	**3420**	**3430**	**3402**	**3380**
3373	**3435**	**3437**	**3411**	**3395**	**3359**
3364	**3428**	**3425**	**3420**	**3390**	**3339**
3355	**3425**	**3415**	**3425**	**3385**	**3322**
3347	**3418**	**3407**	**3394**	**3382**	**3309**
3340	**3401**	**3403**	**3393**	**3381**	**3300**
3335	**3389**	**3385**	**3389**	**3380**	**3295**
3331	**3383**	**3369**	**3383**	**3381**	**3294**
3329	**3368**	**3371**	**3344**	**3318**	**3295**
3329	**3363**	**3358**	**3339**	**3320**	**3299**
3330	**3356**	**3347**	**3340**	**3323**	**3305**
3333	**3357**	**3352**	**3333**	**3325**	**3312**
3333	**3348**	**3354**	**3332**	**3325**	**3317**
3329	**3343**	**3335**	**3343**	**3322**	**3316**
3303	**3318**	**3310**	**3302**	**3300**	**3291**
3241	**3253**	**3250**	**3229**	**3198**	**3228**
3061	**3138**	**3133**	**3138**	**3087**	**3121**
2926	**2962**	**2956**	**2938**	**2957**	**2842**
2667	**2708**	**2708**	**2708**	**2695**	**2659**
2391	**2398**	**2401**	**2374**	**2378**	**2334**
2024	**2040**	**2031**	**2016**	**2005**	**2022**
1571	**1635**	**1634**	**1627**	**1581**	**1612**
1133	**1207**	**1205**	**1199**	**1176**	**1114**
725	**793**	**787**	**793**	**731**	**672**
354	**395**	**381**	**395**	**370**	**293**

S	N(N+1)3	N(N+1)5	N(N+1)7	N(N+1)9	N(N+1)11	N(N+1)13
11383876	11088900	11383876	11370384	11383876	11009124	10863616
11329956	11080021,78	11303044	11276164	11142244	11022400	10890000
11303044	11097781,78	11276164	11209104	11168964	11048976	10929636
11289600	11120001,78	11282881	11249316	11122225	11062276	10982596
11262736	11111111,11	11262736	11249316	11102224	11048976	11009124
11209104	11066711,11	11182336	11209104	11182336	11022400	10982596
11029041	10905405,44	11009124	10949481	10903204	10896601	10817521
10601536	10514887,11	10575504	10575504	10419984	10233601	10419984
9859600	9359520,444	9834496	9803161	9834496	9511056	9734400
8779369	8559525,444	8779369	8732025	8637721	8755681	8076964
7376656	7132460,444	7344100	7354944	7344100	7273809	7075600
5779216	5718475,111	5745609	5764801	5631129	5645376	5456896
4161600	4085788,444	4157521	4112784	4060225	4016016	4084441
2683044	2469088,444	2673225	2669956	2647129	2502724	2592100
1476225	1288225	1476225	1456849	1437601	1385329	1243225
628849	525625	628849	619369	628849	532900	452929
156025	124609	156025	144400	156025	136900	85849
2304	981,7777778	2304	1521	2304	1369	225
0	53,77777778	0	121	0	2304	625
0	1024	0	0	0	0	121
0	0	0	0	0	0	2304
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	256	0	0
0	0	16	0	16	0	0
0	0	9	0	729	0	0
12544	12544	11664	12544	10000	12544	12544
126736	126736	123904	126736	107584	94864	126736
444889	350858,7778	435600	444889	404496	398161	444889
1073296	939607,1111	1067089	1065024	1067089	1018081	853776
2062096	2010724	2042041	2030625	2019241	2027776	1784896
3441025	3411409	3433609	3396649	3374569	3337929	3341584
5175625	5091040,111	5171076	5121169	5062500	4892944	5121169
7134241	6911641	7134241	7054336	7134241	6948496	6985449
9156676	8761600	9138529	9138529	9138529	8874441	8779369
11029041	10543009	11002489	10929636	11002489	10909809	10381284
12616704	12231340,44	12595401	12602500	12538681	12215025	11840481
13801225	13208378,78	13756681	13734436	13579225	13564489	13198689
14531344	14047504	14531344	14531344	14470416	14508481	13623481
14830201	14814801	14822500	14807104	14638276	14554225	14100025
14730244	14222955,11	14699556	14699556	14516100	14645929	14707225
14318656	14263211,11	14295961	14235529	14115049	13853284	13571856
13727025	13111641	13727025	13623481	13727025	13667809	13645636
13046544	12702096	13039321	12988816	12924025	12638025	12152196
12355225	11794645,44	12327121	12292036	12215025	12159169	11854249
11723776	11596295,11	11703241	11641744	11648569	11323225	10909809
11189025	10927432,11	11155600	11115556	11048976	10975969	11002489
10784656	10368400	10764961	10784656	10660225	10673289	10400625
10517049	10443669,44	10517049	10510564	10517049	10387729	9903609
10374841	10017225	10336225	10349089	10233601	10099684	10265616
10361961	10178226,78	10329796	10291264	10227204	10227204	9828225
10445824	10355524	10400625	10413529	10297681	10323369	10169721
10588516	10506241,78	10588516	10569001	10432900	10381284	10452289
10791225	10651520,11	10758400	10791225	10653696	10439361	10666756

11015761	10778089	10982596	10975969	10929636	10876804	10804369
11235904	10859221,78	11189025	11202409	11029041	10870209	10876804
11431161	10901002,78	11417641	11383876	11363641	11269449	10863616
11594025	11485321	11553201	11539609	11498881	11235904	10797796
11716929	11476285,44	11696400	11682724	11641744	11628100	11553201
11792356	11442433,78	11758041	11703241	11758041	11580409	11431161
11819844	11388375,11	11799225	11716929	11634921	11532816	11296321
11799225	11318738,78	11751184	11730625	11696400	11485321	11155600
11758041	11260498,78	11737476	11662225	11737476	11464996	11042329
11689561	11206872,11	11641744	11614464	11532816	11437924	10949481
11607649	11153373,44	11573604	11573604	11519236	11424400	10890000
11526025	11117778,78	11478544	11451456	11478544	11424400	10850436
11444689	11091120,11	11437924	11343424	11437924	11424400	10837264
11377129	11077802,78	11329956	11356900	11168964	11009124	10850436
11336689	11091120,11	11309769	11289600	11148921	11035684	10890000
11296321	11095561	11262736	11209104	11155600	11035684	10929636
11282881	11108889	11276164	11235904	11115556	11062276	10982596
11262736	11115556	11262736	11256025	11102224	11055625	11002489
11215801	11077802,78	11189025	11115556	11189025	11029041	10995856
11029041	10905405,44	11015761	10949481	10909809	10890000	10824100
10601536	10510564	10582009	10569001	10426441	10233601	10419984
9859600	9359520,444	9834496	9803161	9834496	9511056	9728161
8773444	8553675,111	8767521	8726116	8625969	8749764	8071281
7371225	7127120,111	7371225	7349521	7327849	7273809	7070281
5779216	5721664	5750404	5769604	5635876	5650129	5456896
4165681	4092529	4161600	4120900	4064256	4020025	4092529
2689600	2475377,778	2679769	2676496	2653641	2505889	2601769
1473796	1284444,444	1456849	1452025	1437601	1378276	1240996
628849	523693,4444	628849	616225	628849	534361	451584
156025	125080,1111	156025	145161	156025	136900	84681
2209	920,1111111	2209	1444	2209	1296	225
0	53,77777778	0	121	0	2209	625
0	981,7777778	0	225	0	0	121
0	0	0	0	0	0	2209
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**4**	**0**	**4**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
4	**4**	**4**	**4**	**4**	**4**	**4**
0	**0**	**1**	**0**	**1**	**4**	**0**
0	**1,777777778**	**1**	**4**	**1**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**4**
1	**1**	**4**	**1**	**4**	**1**	**1**
1	**1**	**1**	**1**	**1**	**0**	**1**
0	**0,444444444**	**0**	**1**	**0**	**1**	**0**
9	**5,444444444**	**9**	**4**	**9**	**9**	**4**
0	**0**	**0**	**0**	**0**	**9**	**1**
0	**4**	**0**	**9**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**9**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**

0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**9**	**0**	**9**	**0**	**0**
0	**0**	**49**	**0**	**961**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**
9801	**9801**	**9025**	**9801**	**7569**	**9801**	**9801**
123201	**123201**	**122500**	**123201**	**122500**	**99856**	**123201**
451584	**367236**	**451584**	**447561**	**440896**	**410881**	**451584**
1073296	**946080,4444**	**1065024**	**1042441**	**1048576**	**1008016**	**877969**
2076481	**2015453,444**	**2076481**	**2076481**	**2007889**	**2042041**	**1811716**
3474496	**3444736**	**3459600**	**3429904**	**3400336**	**3352561**	**3356224**
5216656	**5116644**	**5202961**	**5212089**	**5094049**	**5180176**	**5161984**
7182400	**6927424**	**7166329**	**7139584**	**7166329**	**6948496**	**7006609**
9168784	**8737936**	**9138529**	**9096256**	**9138529**	**8832784**	**8737936**
11042329	**10517049**	**11009124**	**10989225**	**11009124**	**10909809**	**10336225**
12638025	**12243001**	**12630916**	**12609601**	**12574116**	**12222016**	**11799225**
13816089	**13213225**	**13793796**	**13734436**	**13616100**	**13557124**	**13198689**
14538969	**14040009**	**14493249**	**14516100**	**14493249**	**14500864**	**13616100**
14814801	**14789152,11**	**14814801**	**14776336**	**14630625**	**14531344**	**14062500**
14707225	**14195312,11**	**14691889**	**14668900**	**14508481**	**14638276**	**14668900**
14303524	**14258176**	**14303524**	**14235529**	**14122564**	**13860729**	**13549761**
13719616	**13118884**	**13675204**	**13638249**	**13498276**	**13675204**	**13653025**
13032100	**12697344,44**	**13032100**	**12988816**	**12916836**	**12630916**	**12159169**
12334144	**11778624**	**12313081**	**12278016**	**12201049**	**12152196**	**11847364**
11710084	**11591755,11**	**11696400**	**11641744**	**11641744**	**11329956**	**10903204**
11182336	**10934044,44**	**11135569**	**11128896**	**11029041**	**10982596**	**11009124**
10778089	**10370546,78**	**10751841**	**10686361**	**10647169**	**10673289**	**10413529**

10510564	10441515,11	10510564	10510564	10510564	10387729	9909904
10387729	10034112,11	10342656	10368400	10240000	10118761	10284849
10368400	10188864	10329796	10304100	10227204	10220809	9840769
10445824	10346944,44	10400625	10400625	10297681	10316944	10176100
10595025	10508402,78	10595025	10569001	10439361	10387729	10445824
10797796	10658048,44	10758400	10797796	10653696	10439361	10666756
11015761	10773712,11	10975969	10969344	10923025	10870209	10804369
11235904	10854828,44	11235904	11195716	11235904	10870209	10870209
11437924	10907607,11	11424400	11390625	11370384	11276164	10863616
11600836	11492100	11566801	11546404	11512449	11235904	10804369
11723776	11478544	11703241	11682724	11648569	11628100	11560000
11792356	11437924	11764900	11696400	11764900	11573604	11424400
11812969	11377129	11799225	11812969	11634921	11526025	11282881
11799225	11318738,78	11751184	11730625	11696400	11492100	11148921
11751184	11258261,78	11730625	11662225	11730625	11458225	11035684
11682724	11200177,78	11682724	11607649	11519236	11437924	10949481
11607649	11157826,78	11566801	11580409	11512449	11431161	10890000
11526025	11122225	11485321	11458225	11485321	11424400	10857025
11451456	11097781,78	11444689	11350161	11444689	11431161	10850436
11383876	11084460,44	11343424	11363641	11182336	11009124	10857025
11329956	11080021,78	11309769	11276164	11148921	11022400	10883401
11296321	11091120,11	11262736	11202409	11155600	11042329	10923025
11276164	11106667,11	11269449	11235904	11108889	11055625	10969344
11256025	11108889	11209104	11249316	11102224	11055625	11002489
11215801	11082241	11175649	11122225	11175649	11035684	10995856
11029041	10909809	11009124	10956100	10903204	10890000	10830681
10595025	10504081	10582009	10562500	10426441	10227204	10419984
9872164	9371761,778	9847044	9815689	9847044	9529569	9740641
8779369	8563426,778	8773444	8737936	8631844	8743849	8076964
7365796	7114667,111	7333264	7333264	7333264	7263025	7070281
5779216	5718475,111	5750404	5764801	5635876	5654884	5447556
4165681	4095226,778	4161600	4124961	4064256	4020025	4088484
2683044	2469088,444	2673225	2669956	2647129	2499561	2598544
1473796	1284444,444	1456849	1452025	1437601	1382976	1240996
628849	525625	628849	619369	628849	534361	451584
156025	125080,1111	156025	145161	156025	136900	85849
1212658261	1179902826	1210081113	1207007930	1200160281	1187208913	1,164E+09

Srms	Nrms(N+1)3	Nrms(N+1)5	Nrms(N+1)7	Nrms(N+1)9	Nrms(N+1)11	Nrms(N+1)13
136,56	134,70	136,42	136,24	135,86	135,12	133,79

BIT's	SNR	dB
2	1,03	11,89
4	1,00	0,92
6	1,00	2,03
8	1,01	4,50
10	1,02	9,21
12	1,04	17,80

Appendix C: Mod-Δ (Gaussian) Data Analysis

#	I	Q	MD(N+1)3		MD(N+1)5	MD(N+1)7	MD(N+1)9	MD(N+1)11	MD(N+1)13
0	3374	3432	2	2	6	14	24	44	34
1	3366	3422	2	2	3	10	10	12	42
2	3362	3410	2	2	0	10	26	36	16
3	3360	3397	2	2	4	5	28	20	48
4	3356	3386	0	0	5	12	31	16	125
5	3348	3376	1	1	4	2	18	28	22371
6	3321	3367	0	0	5	11	1	22	6
7	3256	3362	3	3	7	0	12	12	15
8	3140	3360	1	1	1	8	26	7	65
9	2963	3357	0	0	3	3	23	50	69
10	2716	3351	0	0	6	6	1	40	59
11	2404	3331	2	2	3	0	12	61	80
12	2040	3276	24578	24578	4	6	27	46	84
13	1638	3175	28419	28419	3	10	18	34	89
14	1215	3014	1	1	2941	0	23	37	61
15	793	2784	2	2	1	1	6	35	64
16	395	2487	0	0	3	14	25	29	108
17	48	2135	0	0	6	16390	12	14	126
18	0	1742	1	1	2	15	9	28	124
19	0	1322	2	2	4	15	0	41	9
20	0	897	0	0	0	5	28	32	37
21	0	491	1	1	4	4	15	60	44
22	0	129	0	0	3	8	29	47	22
23	0	0	0	0	6	12	8	29	0
24	0	0	3	3	0	3	18	8	30
25	0	0	1	1	4	14	7	18	7
26	0	0	0	0	6	8	4	7	51
27	0	0	2	2	4	4	13	4	22
28	0	0	3	3	5	6	6	13	101
29	0	0	0	0	0	4	0	6	6
30	0	0	1	1	3	4	0	32	118
31	0	0	2	2	4	8	6	32	124
32	0	0	0	0	0	3	10	6	106
33	0	0	0	0	5	12	4	42	124
34	0	0	2	2	6	8	24	4	106

35	0	0	2	2	4	13	28	56	14
36	0	0	0	0	1	14	3	60	93
37	0	0	0	0	0	12	30	35	71
38	0	0	0	0	4	9	8	30	114
39	0	0	3	3	7	0	20	8	22
40	0	0	2	2	5	12	22	52	9
41	0	0	0	0	0	15	4	22	1
42	0	0	0	0	2	13	5	4	40
43	0	0	2	2	7	8	8	4	52
44	0	0	0	0	4	2	3	8	112
45	0	0	1	1	5	7	12	35	97
46	0	0	0	0	6	4	24	12	87
47	0	0	3	3	0	13	13	24	1
48	0	0	0	0	0	6	14	45	83
49	0	0	0	0	6	0	28	14	115
50	0	0	1	1	2	0	9	28	76
51	0	0	2	2	4	6	0	41	9
52	0	0	0	0	0	10	28	32	37
53	0	0	1	1	4	4	15	60	44
54	0	0	0	0	3	8	29	47	22
55	0	0	0	0	6	12	8	29	0
56	0	0	3	3	0	3	18	8	30
57	0	0	1	1	4	14	7	18	30
58	0	0	0	0	6	8	4	55	51
59	112	0	2	2	4	4	29	40	22
60	356	0	2	2	1	6	10	40	85
61	667	0	0	0	3	4	27	18	106
62	1036	0	1	1	7	5	12	60	17
63	1436	0	1	1	0	8	2	31	8
64	1855	61	16307	16307	7	7	9	41	6
65	2275	289	3	3	0	7	7	25	59
66	2671	583	0	0	5	4	7	22	77
67	3026	940	3	3	6	9	14	49	125
68	3321	1334	0	0	2	31117	28	28	47
69	3552	1749	32575	32575	0	15	30	38	22368
70	3715	2172	32740	32740	7	8	11	2	16466
71	3812	2576	26354	26354	3	2	31192	19	25
72	3851	2942	0	0	32752	5	1	50	22253
73	3838	3253	32468	32468	6	15	2	30	12
74	3784	3501	1	1	2	0	16396	61	38
75	3705	3682	32566	32566	6136	29628	1	33	124
76	3612	3794	32743	32743	0	13	31	3	6009
77	3515	3847	1	1	0	6021	7	3	31613

78	3424	3846	1	1	6	8540	24	29	18
79	3345	3801	3	3	1	6	30	44	97
80	3284	3727	3	3	4	16642	2	24	100
81	3243	3636	1	1	1	11	7	35	71
82	3221	3539	0	0	7	0	30	47	119
83	3219	3445	2	2	7	7	19	9	5822
84	3232	3363	2	2	0	14	28	22	56
85	3254	3297	2	2	2	15	5	17	204
86	3285	3251	3	3	0	13	18	38	76
87	3319	3225	0	0	5	15	31	53	85
88	3352	3218	0	0	0	3	10	61	21
89	3381	3228	2	2	1	4	28	31	31
90	3405	3248	3	3	3	13	17	38	104
91	3423	3276	0	0	3	11	12	46	99
92	3434	3310	1	1	6	14	16	59	68
93	3438	3344	3	3	6	9	14	49	112
94	3435	3374	2	2	6	2	11	5	100
95	3429	3400	16641	16641	1	7	11	59	103
96	3419	3419	3	3	3	13	5	21	79
97	3407	3432	3	3	4	10	19	45	87
98	3395	3437	1	1	1	3	27	59	57
99	3383	3437	3	3	3	2	19	37	81
100	3373	3432	3	3	6	9	16	35	20
101	3367	3422	1	1	7	11	5	4	21876
102	3361	3410	3	3	5	12	9	61	25
103	3359	3398	3	3	6	7	19	36	55
104	3356	3386	3	3	1	9	18	9	40
105	3349	3375	1	1	5	6	25	15	29
106	3321	3368	0	0	3	14	30	60	61
107	3256	3362	2	2	7	7	0	9	45
108	3140	3359	2	2	0	14	7	26	40
109	2962	3357	3	3	7	12	30	62	37
110	2715	3351	0	0	16385	13	19	48	105
111	2404	3331	0	0	4	5	17	17	28
112	2041	3276	1576	1576	1	10	7	21	55
113	1640	3175	2	2	6	2	4	12	108
114	1214	3013	2	2	0	11	7	53	52
115	793	2783	16385	16385	5	10	25	52	71
116	395	2487	3	3	3	3	7	15	62
117	47	2136	1	1	3	12	30	60	55
118	0	1743	0	0	3	6	29	47	69
119	0	1322	0	0	6	5	8	29	0
120	0	897	3	3	0	14	18	8	30

121	0	491	1	1	4	13	7	18	7
122	0	128	0	0	6	8	4	7	51
123	0	0	2	2	4	4	13	4	22
124	0	0	3	3	5	6	6	13	101
125	0	0	0	0	0	4	0	6	6
126	0	0	1	1	3	4	0	32	118
127	0	0	2	2	4	8	6	32	124
128	0	0	0	0	0	3	10	6	106
129	0	0	0	0	5	12	4	42	124
130	0	0	2	2	6	8	24	4	106
131	0	0	2	2	4	13	28	56	14
132	0	0	0	0	1	14	3	60	93
133	0	0	0	0	0	12	30	35	71
134	0	0	0	0	4	9	8	30	114
135	0	0	3	3	7	0	20	8	22
136	0	0	2	2	5	12	22	52	9
137	0	0	2	2	0	15	4	24	1
138	2	0	0	0	4	13	7	4	40
139	0	0	2	2	7	8	8	4	54
140	0	0	0	0	4	2	3	8	112
141	0	0	2	2	5	7	12	36	97
142	1	0	1	1	7	6	25	13	87
143	1	2	3	3	1	13	14	24	2
144	0	0	3	3	0	6	14	48	84
145	3	0	0	0	1	0	31	14	115
146	0	0	1	1	2	1	9	28	79
147	0	0	2	2	4	7	0	41	9
148	0	1	0	0	0	10	28	32	37
149	0	0	1	1	4	7	15	60	44
150	0	2	0	0	3	8	29	47	22
151	0	1	0	0	6	12	8	29	0
152	0	0	3	3	0	3	18	8	30
153	0	0	1	1	4	14	7	18	30
154	0	0	0	0	6	8	4	7	51
155	0	0	2	2	4	4	13	4	22
156	0	0	3	3	5	6	6	13	101
157	0	0	0	0	0	4	0	6	6
158	0	0	1	1	3	5	0	32	118
159	0	0	2	2	4	8	6	32	124
160	0	0	0	0	0	3	10	6	106
161	0	0	0	0	5	12	4	42	124
162	0	0	2	2	6	8	24	4	106
163	0	0	2	2	4	13	28	56	14

164	**0**	**0**	**0**	**0**	**1**	**14**	**3**	**60**	**93**
165	**0**	**0**	**0**	**0**	**0**	**12**	**30**	**35**	**71**
166	**0**	**0**	**0**	**0**	**4**	**9**	**8**	**30**	**114**
167	**0**	**0**	**3**	**3**	**7**	**0**	**20**	**8**	**22**
168	**0**	**0**	**2**	**2**	**5**	**12**	**22**	**52**	**9**
169	**0**	**0**	**0**	**0**	**0**	**15**	**4**	**22**	**1**
170	**0**	**0**	**0**	**0**	**2**	**13**	**4**	**4**	**40**
171	**0**	**0**	**2**	**2**	**7**	**8**	**8**	**5**	**52**
172	**0**	**0**	**0**	**0**	**4**	**2**	**3**	**8**	**112**
173	**0**	**0**	**1**	**1**	**5**	**7**	**12**	**35**	**97**
174	**0**	**0**	**0**	**0**	**6**	**4**	**24**	**12**	**87**
175	**0**	**0**	**3**	**3**	**0**	**13**	**13**	**24**	**1**
176	**0**	**0**	**0**	**0**	**0**	**6**	**14**	**45**	**83**
177	**0**	**0**	**0**	**0**	**6**	**0**	**28**	**14**	**115**
178	**0**	**0**	**1**	**1**	**2**	**0**	**9**	**28**	**76**
179	**0**	**0**	**2**	**2**	**4**	**6**	**0**	**41**	**9**
180	**0**	**0**	**0**	**0**	**0**	**10**	**28**	**32**	**37**
181	**0**	**0**	**1**	**1**	**4**	**4**	**15**	**60**	**44**
182	**0**	**0**	**0**	**0**	**3**	**8**	**29**	**47**	**22**
183	**0**	**0**	**0**	**0**	**6**	**12**	**8**	**29**	**0**
184	**0**	**0**	**3**	**3**	**0**	**3**	**18**	**8**	**30**
185	**0**	**0**	**1**	**1**	**4**	**14**	**7**	**18**	**7**
186	**0**	**0**	**0**	**0**	**6**	**8**	**4**	**7**	**51**
187	**0**	**0**	**2**	**2**	**4**	**4**	**13**	**4**	**22**
188	**0**	**0**	**3**	**3**	**4**	**6**	**6**	**13**	**101**
189	**0**	**0**	**0**	**0**	**0**	**4**	**0**	**6**	**6**
190	**0**	**0**	**1**	**1**	**3**	**5**	**0**	**32**	**118**
191	**0**	**0**	**2**	**2**	**4**	**8**	**6**	**32**	**124**
192	**0**	**0**	**0**	**0**	**0**	**3**	**10**	**6**	**106**
193	**0**	**0**	**0**	**0**	**5**	**12**	**4**	**42**	**124**
194	**0**	**0**	**2**	**2**	**6**	**8**	**24**	**4**	**106**
195	**0**	**0**	**2**	**2**	**4**	**13**	**28**	**56**	**14**
196	**0**	**0**	**3**	**3**	**1**	**14**	**3**	**31**	**93**
197	**99**	**0**	**3**	**3**	**3**	**12**	**1**	**2**	**71**
198	**351**	**0**	**0**	**0**	**3**	**9**	**7**	**62**	**85**
199	**672**	**0**	**3**	**3**	**7**	**0**	**20**	**20**	**117**
200	**1036**	**0**	**3**	**3**	**1**	**12**	**2**	**21**	**41**
201	**1441**	**0**	**0**	**0**	**1**	**2**	**5**	**30**	**13**
202	**1864**	**50**	**2616**	**2616**	**2**	**12**	**13**	**48**	**73**
203	**2284**	**280**	**2**	**2**	**3**	**8**	**20**	**61**	**124**
204	**2680**	**587**	**0**	**0**	**4**	**14**	**27**	**28**	**92**
205	**3028**	**941**	**3**	**3**	**1**	**8**	**0**	**30**	**89**
206	**3323**	**1337**	**3**	**3**	**1**	**28428**	**19**	**47**	**43**

207	3555	1758	0	0	3	9	16	29	124
208	3717	2181	32565	32565	4093	14	19	18	54
209	3813	2585	1	1	3	4	32513	23	22264
210	3849	2946	0	0	30651	11	18	23	49
211	3835	3255	32600	32600	10167	9	31131	16431	530
212	3782	3504	32760	32760	6	15	2	24	32
213	3704	3682	32743	32743	10124	25529	7	22	114
214	3610	3795	0	0	5	1	23	39	5950
215	3512	3845	2	2	6	7	0	59	26
216	3422	3842	3	3	6	31673	16	24	86
217	3344	3798	0	0	4	6	23	37	101
218	3283	3725	2	2	1	32642	23	49	67
219	3242	3634	1	1	6	12	23	27	105
220	3223	3536	3	3	4	4	29	33	15
221	3220	3443	0	0	4	4	20	38	29
222	3232	3362	0	0	3	7	0	23	10
223	3255	3296	0	0	3	2	29	54	28
224	3286	3250	3	3	6	10	0	61	33
225	3319	3226	0	0	4	0	27	2	82
226	3352	3219	0	0	6	8	16	58	16865
227	3382	3228	16384	16384	2	4	18	6	38
228	3406	3248	0	0	7	4	17	28	19
229	3424	3278	2	2	0	3	30	13	1205
230	3434	3311	1	1	6	1	18	11	82
231	3437	3343	2	2	4	6	1	51	0
232	3435	3375	2	2	0	10	1	24	118
233	3428	3400	2	2	4	15	8	48	108
234	3418	3420	3	3	4	7	31	19	12
235	3407	3432	1	1	6	5	23	7	14
236	3395	3437	0	0	7	13	6	0	63
237	3384	3437	3	3	5	11	4	17	36
238	3374	3430	2	2	4	14	6	50	15
239	3366	3421	0	0	6	12	19	57	47
240	3361	3410	2	2	1	9	15	11	121
241	3358	3398	3	3	4	8	26	41	6004
242	3355	3387	2	2	5	14	4	49	106
243	3349	3376	3	3	1	12	21	34	30244
244	3321	3367	3	3	1	11	21	23	58
245	3255	3362	3	3	3	2	6	2	16421
246	3142	3359	3	3	1	3	3	2	77
247	2963	3356	2	2	1	1	27	55	70
248	2714	3352	1983	1983	2	12	12	44	49
249	2404	3331	2	2	0	5	11	11	56

250	2041	3275	2	2	7	14	29	45	23
251	1638	3176	32640	32640	2	7	19	2	15
252	1214	3015	27904	27904	3	0	16388	38	75
253	793	2782	3	3	1	8	25	17	68
254	395	2486	0	0	6	14	11	15	15

Noise(N+1)3	Noise(N+1)5	Noise(N+1)7	Noise(N+1)9	Noise(N+1)11	Noise(N+1)13
3372	3368	3360	3350	3330	3340
3364	3363	3356	3356	3354	3324
3360	3362	3352	3336	3326	3346
3358	3356	3355	3332	3340	3312
3356	3351	3344	3325	3340	3231
3347	3344	3346	3330	3320	-19023
3321	3316	3310	3320	3299	3315
3253	3249	3256	3244	3244	3241
3139	3139	3132	3114	3133	3075
2963	2960	2960	2940	2913	2894
2716	2710	2710	2715	2676	2657
2402	2401	2404	2392	2343	2324
-22538	2036	2034	2013	1994	1956
-26781	1635	1628	1620	1604	1549
1214	-1726	1215	1192	1178	1154
791	792	792	787	758	729
395	392	381	370	366	287
48	42	-16342	36	34	-78
-1	-2	-15	-9	-28	-124
-2	-4	-15	0	-41	-9
0	0	-5	-28	-32	-37
-1	-4	-4	-15	-60	-44
0	-3	-8	-29	-47	-22
0	-6	-12	-8	-29	0
-3	0	-3	-18	-8	-30
-1	-4	-14	-7	-18	-7
0	-6	-8	-4	-7	-51
-2	-4	-4	-13	-4	-22
-3	-5	-6	-6	-13	-101
0	0	-4	0	-6	-6
-1	-3	-4	0	-32	-118
-2	-4	-8	-6	-32	-124
0	0	-3	-10	-6	-106

0	**-5**	**-12**	**-4**	**-42**	**-124**
-2	**-6**	**-8**	**-24**	**-4**	**-106**
-2	**-4**	**-13**	**-28**	**-56**	**-14**
0	**-1**	**-14**	**-3**	**-60**	**-93**
0	**0**	**-12**	**-30**	**-35**	**-71**
0	**-4**	**-9**	**-8**	**-30**	**-114**
-3	**-7**	**0**	**-20**	**-8**	**-22**
-2	**-5**	**-12**	**-22**	**-52**	**-9**
0	**0**	**-15**	**-4**	**-22**	**-1**
0	**-2**	**-13**	**-5**	**-4**	**-40**
-2	**-7**	**-8**	**-8**	**-4**	**-52**
0	**-4**	**-2**	**-3**	**-8**	**-112**
-1	**-5**	**-7**	**-12**	**-35**	**-97**
0	**-6**	**-4**	**-24**	**-12**	**-87**
-3	**0**	**-13**	**-13**	**-24**	**-1**
0	**0**	**-6**	**-14**	**-45**	**-83**
0	**-6**	**0**	**-28**	**-14**	**-115**
-1	**-2**	**0**	**-9**	**-28**	**-76**
-2	**-4**	**-6**	**0**	**-41**	**-9**
0	**0**	**-10**	**-28**	**-32**	**-37**
-1	**-4**	**-4**	**-15**	**-60**	**-44**
0	**-3**	**-8**	**-29**	**-47**	**-22**
0	**-6**	**-12**	**-8**	**-29**	**0**
-3	**0**	**-3**	**-18**	**-8**	**-30**
-1	**-4**	**-14**	**-7**	**-18**	**-30**
0	**-6**	**-8**	**-4**	**-55**	**-51**
110	**108**	**108**	**83**	**72**	**90**
354	**355**	**350**	**346**	**316**	**271**
667	**664**	**663**	**640**	**649**	**561**
1035	**1029**	**1031**	**1024**	**976**	**1019**
1435	**1436**	**1428**	**1434**	**1405**	**1428**
-14452	**1848**	**1848**	**1846**	**1814**	**1849**
2272	**2275**	**2268**	**2268**	**2250**	**2216**
2671	**2666**	**2667**	**2664**	**2649**	**2594**
3023	**3020**	**3017**	**3012**	**2977**	**2901**
3321	**3319**	**-27796**	**3293**	**3293**	**3274**
-29023	**3552**	**3537**	**3522**	**3514**	**-18816**
-29025	**3708**	**3707**	**3704**	**3713**	**-12751**
-22542	**3809**	**3810**	**-27380**	**3793**	**3787**
3851	**-28901**	**3846**	**3850**	**3801**	**-18402**
-28630	**3832**	**3823**	**3836**	**3808**	**3826**
3783	**3782**	**3784**	**-12612**	**3723**	**3746**
-28861	**-2431**	**-25923**	**3704**	**3672**	**3581**
-29131	**3612**	**3599**	**3581**	**3609**	**-2397**

3514	**3515**	**-2506**	**3508**	**3512**	**-28098**
3423	**3418**	**-5116**	**3400**	**3395**	**3406**
3342	**3344**	**3339**	**3315**	**3301**	**3248**
3281	**3280**	**-13358**	**3282**	**3260**	**3184**
3242	**3242**	**3232**	**3236**	**3208**	**3172**
3221	**3214**	**3221**	**3191**	**3174**	**3102**
3217	**3212**	**3212**	**3200**	**3210**	**-2603**
3230	**3232**	**3218**	**3204**	**3210**	**3176**
3252	**3252**	**3239**	**3249**	**3237**	**3050**
3282	**3285**	**3272**	**3267**	**3247**	**3209**
3319	**3314**	**3304**	**3288**	**3266**	**3234**
3352	**3352**	**3349**	**3342**	**3291**	**3331**
3379	**3380**	**3377**	**3353**	**3350**	**3350**
3402	**3402**	**3392**	**3388**	**3367**	**3301**
3423	**3420**	**3412**	**3411**	**3377**	**3324**
3433	**3428**	**3420**	**3418**	**3375**	**3366**
3435	**3432**	**3429**	**3424**	**3389**	**3326**
3433	**3429**	**3433**	**3424**	**3430**	**3335**
-13212	**3428**	**3422**	**3418**	**3370**	**3326**
3416	**3416**	**3406**	**3414**	**3398**	**3340**
3404	**3403**	**3397**	**3388**	**3362**	**3320**
3394	**3394**	**3392**	**3368**	**3336**	**3338**
3380	**3380**	**3381**	**3364**	**3346**	**3302**
3370	**3367**	**3364**	**3357**	**3338**	**3353**
3366	**3360**	**3356**	**3362**	**3363**	**-18509**
3358	**3356**	**3349**	**3352**	**3300**	**3336**
3356	**3353**	**3352**	**3340**	**3323**	**3304**
3353	**3355**	**3347**	**3338**	**3347**	**3316**
3348	**3344**	**3343**	**3324**	**3334**	**3320**
3321	**3318**	**3307**	**3291**	**3261**	**3260**
3254	**3249**	**3249**	**3256**	**3247**	**3211**
3138	**3140**	**3126**	**3133**	**3114**	**3100**
2959	**2955**	**2950**	**2932**	**2900**	**2925**
2715	**-13670**	**2702**	**2696**	**2667**	**2610**
2404	**2400**	**2399**	**2387**	**2387**	**2376**
465	**2040**	**2031**	**2034**	**2020**	**1986**
1638	**1634**	**1638**	**1636**	**1628**	**1532**
1212	**1214**	**1203**	**1207**	**1161**	**1162**
-15592	**788**	**783**	**768**	**741**	**722**
392	**392**	**392**	**388**	**380**	**333**
46	**44**	**35**	**17**	**-13**	**-8**
0	**-3**	**-6**	**-29**	**-47**	**-69**
0	**-6**	**-5**	**-8**	**-29**	**0**
-3	**0**	**-14**	**-18**	**-8**	**-30**

-1	-4	-13	-7	-18	-7
0	-6	-8	-4	-7	-51
-2	-4	-4	-13	-4	-22
-3	-5	-6	-6	-13	-101
0	0	-4	0	-6	-6
-1	-3	-4	0	-32	-118
-2	-4	-8	-6	-32	-124
0	0	-3	-10	-6	-106
0	-5	-12	-4	-42	-124
-2	-6	-8	-24	-4	-106
-2	-4	-13	-28	-56	-14
0	-1	-14	-3	-60	-93
0	0	-12	-30	-35	-71
0	-4	-9	-8	-30	-114
-3	-7	0	-20	-8	-22
-2	-5	-12	-22	-52	-9
-2	0	-15	-4	-24	-1
2	-2	-11	-5	-2	-38
-2	-7	-8	-8	-4	-54
0	-4	-2	-3	-8	-112
-2	-5	-7	-12	-36	-97
0	-6	-5	-24	-12	-86
-2	0	-12	-13	-23	-1
-3	0	-6	-14	-48	-84
3	2	3	-28	-11	-112
-1	-2	-1	-9	-28	-79
-2	-4	-7	0	-41	-9
0	0	-10	-28	-32	-37
-1	-4	-7	-15	-60	-44
0	-3	-8	-29	-47	-22
0	-6	-12	-8	-29	0
-3	0	-3	-18	-8	-30
-1	-4	-14	-7	-18	-30
0	-6	-8	-4	-7	-51
-2	-4	-4	-13	-4	-22
-3	-5	-6	-6	-13	-101
0	0	-4	0	-6	-6
-1	-3	-5	0	-32	-118
-2	-4	-8	-6	-32	-124
0	0	-3	-10	-6	-106
0	-5	-12	-4	-42	-124
-2	-6	-8	-24	-4	-106
-2	-4	-13	-28	-56	-14
0	-1	-14	-3	-60	-93

0	**0**	**-12**	**-30**	**-35**	**-71**
0	**-4**	**-9**	**-8**	**-30**	**-114**
-3	**-7**	**0**	**-20**	**-8**	**-22**
-2	**-5**	**-12**	**-22**	**-52**	**-9**
0	**0**	**-15**	**-4**	**-22**	**-1**
0	**-2**	**-13**	**-4**	**-4**	**-40**
-2	**-7**	**-8**	**-8**	**-5**	**-52**
0	**-4**	**-2**	**-3**	**-8**	**-112**
-1	**-5**	**-7**	**-12**	**-35**	**-97**
0	**-6**	**-4**	**-24**	**-12**	**-87**
-3	**0**	**-13**	**-13**	**-24**	**-1**
0	**0**	**-6**	**-14**	**-45**	**-83**
0	**-6**	**0**	**-28**	**-14**	**-115**
-1	**-2**	**0**	**-9**	**-28**	**-76**
-2	**-4**	**-6**	**0**	**-41**	**-9**
0	**0**	**-10**	**-28**	**-32**	**-37**
-1	**-4**	**-4**	**-15**	**-60**	**-44**
0	**-3**	**-8**	**-29**	**-47**	**-22**
0	**-6**	**-12**	**-8**	**-29**	**0**
-3	**0**	**-3**	**-18**	**-8**	**-30**
-1	**-4**	**-14**	**-7**	**-18**	**-7**
0	**-6**	**-8**	**-4**	**-7**	**-51**
-2	**-4**	**-4**	**-13**	**-4**	**-22**
-3	**-4**	**-6**	**-6**	**-13**	**-101**
0	**0**	**-4**	**0**	**-6**	**-6**
-1	**-3**	**-5**	**0**	**-32**	**-118**
-2	**-4**	**-8**	**-6**	**-32**	**-124**
0	**0**	**-3**	**-10**	**-6**	**-106**
0	**-5**	**-12**	**-4**	**-42**	**-124**
-2	**-6**	**-8**	**-24**	**-4**	**-106**
-2	**-4**	**-13**	**-28**	**-56**	**-14**
-3	**-1**	**-14**	**-3**	**-31**	**-93**
96	**96**	**87**	**98**	**97**	**28**
351	**348**	**342**	**344**	**289**	**266**
669	**665**	**672**	**652**	**652**	**555**
1033	**1035**	**1024**	**1034**	**1015**	**995**
1441	**1440**	**1439**	**1436**	**1411**	**1428**
-752	**1862**	**1852**	**1851**	**1816**	**1791**
2282	**2281**	**2276**	**2264**	**2223**	**2160**
2680	**2676**	**2666**	**2653**	**2652**	**2588**
3025	**3027**	**3020**	**3028**	**2998**	**2939**
3320	**3322**	**-25105**	**3304**	**3276**	**3280**
3555	**3552**	**3546**	**3539**	**3526**	**3431**
-28848	**-376**	**3703**	**3698**	**3699**	**3663**

3812	3810	3809	-28700	3790	-18451
3849	-26802	3838	3831	3826	3800
-28765	-6332	3826	-27296	-12596	3305
-28978	3776	3767	3780	3758	3750
-29039	-6420	-21825	3697	3682	3590
3610	3605	3609	3587	3571	-2340
3510	3506	3505	3512	3453	3486
3419	3416	-28251	3406	3398	3336
3344	3340	3338	3321	3307	3243
3281	3282	-29359	3260	3234	3216
3241	3236	3230	3219	3215	3137
3220	3219	3219	3194	3190	3208
3220	3216	3216	3200	3182	3191
3232	3229	3225	3232	3209	3222
3255	3252	3253	3226	3201	3227
3283	3280	3276	3286	3225	3253
3319	3315	3319	3292	3317	3237
3352	3346	3344	3336	3294	-13513
-13002	3380	3378	3364	3376	3344
3406	3399	3402	3389	3378	3387
3422	3424	3421	3394	3411	2219
3433	3428	3433	3416	3423	3352
3435	3433	3431	3436	3386	3437
3433	3435	3425	3434	3411	3317
3426	3424	3413	3420	3380	3320
3415	3414	3411	3387	3399	3406
3406	3401	3402	3384	3400	3393
3395	3388	3382	3389	3395	3332
3381	3379	3373	3380	3367	3348
3372	3370	3360	3368	3324	3359
3366	3360	3354	3347	3309	3319
3359	3360	3352	3346	3350	3240
3355	3354	3350	3332	3317	-2646
3353	3350	3341	3351	3306	3249
3346	3348	3337	3328	3315	-26895
3318	3320	3310	3300	3298	3263
3252	3252	3253	3249	3253	-13166
3139	3141	3139	3139	3140	3065
2961	2962	2962	2936	2908	2893
731	2712	2702	2702	2670	2665
2402	2404	2399	2393	2393	2348
2039	2034	2027	2012	1996	2018
-31002	1636	1631	1619	1636	1623
-26690	1211	1214	-15174	1176	1139

790	792	785	768	776	725
395	389	381	384	380	380

S	N(N+1)3	N(N+1)5	N(N+1)7	N(N+1)9	N(N+1)11	N(N+1)13
11383876	11370384	11343424	11289600	11222500	11088900	11155600
11329956	11316496	11309769	11262736	11262736	11249316	11048976
11303044	11289600	11303044	11235904	11128896	11062276	11195716
11289600	11276164	11262736	11256025	11102224	11155600	10969344
11262736	11262736	11229201	11182336	11055625	11155600	10439361
11209104	11202409	11182336	11195716	11088900	11022400	361874529
11029041	11029041	10995856	10956100	11022400	10883401	10989225
10601536	10582009	10556001	10601536	10523536	10523536	10504081
9859600	9853321	9853321	9809424	9696996	9815689	9455625
8779369	8779369	8761600	8761600	8643600	8485569	8375236
7376656	7376656	7344100	7344100	7371225	7160976	7059649
5779216	5769604	5764801	5779216	5721664	5489649	5400976
4161600	507961444	4145296	4137156	4052169	3976036	3825936
2683044	717221961	2673225	2650384	2624400	2572816	2399401
1476225	1473796	2979076	1476225	1420864	1387684	1331716
628849	625681	627264	627264	619369	574564	531441
156025	156025	153664	145161	136900	133956	82369
2304	2304	1764	267060964	1296	1156	6084
0	1	4	225	81	784	15376
0	4	16	225	0	1681	81
0	0	0	25	784	1024	1369
0	1	16	16	225	3600	1936
0	0	9	64	841	2209	484
0	0	36	144	64	841	0
0	9	0	9	324	64	900
0	1	16	196	49	324	49
0	0	36	64	16	49	2601
0	4	16	16	169	16	484
0	9	25	36	36	169	10201
0	0	0	16	0	36	36
0	1	9	16	0	1024	13924
0	4	16	64	36	1024	15376
0	0	0	9	100	36	11236
0	0	25	144	16	1764	15376
0	4	36	64	576	16	11236
0	4	16	169	784	3136	196
0	0	1	196	9	3600	8649
0	0	0	144	900	1225	5041

0	0	16	81	64	900	12996
0	9	49	0	400	64	484
0	4	25	144	484	2704	81
0	0	0	225	16	484	1
0	0	4	169	25	16	1600
0	4	49	64	64	16	2704
0	0	16	4	9	64	12544
0	1	25	49	144	1225	9409
0	0	36	16	576	144	7569
0	9	0	169	169	576	1
0	0	0	36	196	2025	6889
0	0	36	0	784	196	13225
0	1	4	0	81	784	5776
0	4	16	36	0	1681	81
0	0	0	100	784	1024	1369
0	1	16	16	225	3600	1936
0	0	9	64	841	2209	484
0	0	36	144	64	841	0
0	9	0	9	324	64	900
0	1	16	196	49	324	900
0	0	36	64	16	3025	2601
12544	12100	11664	11664	6889	5184	8100
126736	125316	126025	122500	119716	99856	73441
444889	444889	440896	439569	409600	421201	314721
1073296	1071225	1058841	1062961	1048576	952576	1038361
2062096	2059225	2062096	2039184	2056356	1974025	2039184
3441025	208860304	3415104	3415104	3407716	3290596	3418801
5175625	5161984	5175625	5143824	5143824	5062500	4910656
7134241	7134241	7107556	7112889	7096896	7017201	6728836
9156676	9138529	9120400	9102289	9072144	8862529	8415801
11029041	11029041	11015761	772617616	10843849	10843849	10719076
12616704	842334529	12616704	12510369	12404484	12348196	354041856
13801225	842450625	13749264	13741849	13719616	13786369	162588001
14531344	508141764	14508481	14516100	749664400	14386849	14341369
14830201	14830201	835267801	14791716	14822500	14447601	338633604
14730244	819676900	14684224	14615329	14714896	14500864	14638276
14318656	14311089	14303524	14318656	159062544	13860729	14032516
13727025	832957321	5909761	672001929	13719616	13483584	12823561
13046544	848615161	13046544	12952801	12823561	13024881	5745609
12355225	12348196	12355225	6280036	12306064	12334144	789497604
11723776	11716929	11682724	26173456	11560000	11526025	11600836
11189025	11168964	11182336	11148921	10989225	10896601	10549504
10784656	10764961	10758400	178436164	10771524	10627600	10137856
10517049	10510564	10510564	10445824	10471696	10291264	10061584

10374841	**10374841**	**10329796**	**10374841**	**10182481**	**10074276**	**9622404**
10361961	**10349089**	**10316944**	**10316944**	**10240000**	**10304100**	**6775609**
10445824	**10432900**	**10445824**	**10355524**	**10265616**	**10304100**	**10086976**
10588516	**10575504**	**10575504**	**10491121**	**10556001**	**10478169**	**9302500**
10791225	**10771524**	**10791225**	**10705984**	**10673289**	**10543009**	**10297681**
11015761	**11015761**	**10982596**	**10916416**	**10810944**	**10666756**	**10458756**
11235904	**11235904**	**11235904**	**11215801**	**11168964**	**10830681**	**11095561**
11431161	**11417641**	**11424400**	**11404129**	**11242609**	**11222500**	**11222500**
11594025	**11573604**	**11573604**	**11505664**	**11478544**	**11336689**	**10896601**
11716929	**11716929**	**11696400**	**11641744**	**11634921**	**11404129**	**11048976**
11792356	**11785489**	**11751184**	**11696400**	**11682724**	**11390625**	**11329956**
11819844	**11799225**	**11778624**	**11758041**	**11723776**	**11485321**	**11062276**
11799225	**11785489**	**11758041**	**11785489**	**11723776**	**11764900**	**11122225**
11758041	**174556944**	**11751184**	**11710084**	**11682724**	**11356900**	**11062276**
11689561	**11669056**	**11669056**	**11600836**	**11655396**	**11546404**	**11155600**
11607649	**11587216**	**11580409**	**11539609**	**11478544**	**11303044**	**11022400**
11526025	**11519236**	**11519236**	**11505664**	**11343424**	**11128896**	**11142244**
11444689	**11424400**	**11424400**	**11431161**	**11316496**	**11195716**	**10903204**
11377129	**11356900**	**11336689**	**11316496**	**11269449**	**11142244**	**11242609**
11336689	**11329956**	**11289600**	**11262736**	**11303044**	**11309769**	**342583081**
11296321	**11276164**	**11262736**	**11215801**	**11235904**	**10890000**	**11128896**
11282881	**11262736**	**11242609**	**11235904**	**11155600**	**11042329**	**10916416**
11262736	**11242609**	**11256025**	**11202409**	**11142244**	**11202409**	**10995856**
11215801	**11209104**	**11182336**	**11175649**	**11048976**	**11115556**	**11022400**
11029041	**11029041**	**11009124**	**10936249**	**10830681**	**10634121**	**10627600**
10601536	**10588516**	**10556001**	**10556001**	**10601536**	**10543009**	**10310521**
9859600	**9847044**	**9859600**	**9771876**	**9815689**	**9696996**	**9610000**
8773444	**8755681**	**8732025**	**8702500**	**8596624**	**8410000**	**8555625**
7371225	**7371225**	**186868900**	**7300804**	**7268416**	**7112889**	**6812100**
5779216	**5779216**	**5760000**	**5755201**	**5697769**	**5697769**	**5645376**
4165681	**216225**	**4161600**	**4124961**	**4137156**	**4080400**	**3944196**
2689600	**2683044**	**2669956**	**2683044**	**2676496**	**2650384**	**2347024**
1473796	**1468944**	**1473796**	**1447209**	**1456849**	**1347921**	**1350244**
628849	**243110464**	**620944**	**613089**	**589824**	**549081**	**521284**
156025	**153664**	**153664**	**153664**	**150544**	**144400**	**110889**
2209	**2116**	**1936**	**1225**	**289**	**169**	**64**
0	**0**	**9**	**36**	**841**	**2209**	**4761**
0	**0**	**36**	**25**	**64**	**841**	**0**
0	**9**	**0**	**196**	**324**	**64**	**900**
0	**1**	**16**	**169**	**49**	**324**	**49**
0	**0**	**36**	**64**	**16**	**49**	**2601**
0	**4**	**16**	**16**	**169**	**16**	**484**
0	**9**	**25**	**36**	**36**	**169**	**10201**
0	**0**	**0**	**16**	**0**	**36**	**36**

0	1	9	16	0	1024	13924
0	4	16	64	36	1024	15376
0	0	0	9	100	36	11236
0	0	25	144	16	1764	15376
0	4	36	64	576	16	11236
0	4	16	169	784	3136	196
0	0	1	196	9	3600	8649
0	0	0	144	900	1225	5041
0	0	16	81	64	900	12996
0	9	49	0	400	64	484
0	4	25	144	484	2704	81
0	4	0	225	16	576	1
4	4	4	121	25	4	1444
0	4	49	64	64	16	2916
0	0	16	4	9	64	12544
0	4	25	49	144	1296	9409
1	0	36	25	576	144	7396
1	4	0	144	169	529	1
0	9	0	36	196	2304	7056
9	9	4	9	784	121	12544
0	1	4	1	81	784	6241
0	4	16	49	0	1681	81
0	0	0	100	784	1024	1369
0	1	16	49	225	3600	1936
0	0	9	64	841	2209	484
0	0	36	144	64	841	0
0	9	0	9	324	64	900
0	1	16	196	49	324	900
0	0	36	64	16	49	2601
0	4	16	16	169	16	484
0	9	25	36	36	169	10201
0	0	0	16	0	36	36
0	1	9	25	0	1024	13924
0	4	16	64	36	1024	15376
0	0	0	9	100	36	11236
0	0	25	144	16	1764	15376
0	4	36	64	576	16	11236
0	4	16	169	784	3136	196
0	0	1	196	9	3600	8649
0	0	0	144	900	1225	5041
0	0	16	81	64	900	12996
0	9	49	0	400	64	484
0	4	25	144	484	2704	81
0	0	0	225	16	484	1

0	0	4	169	16	16	1600
0	4	49	64	64	25	2704
0	0	16	4	9	64	12544
0	1	25	49	144	1225	9409
0	0	36	16	576	144	7569
0	9	0	169	169	576	1
0	0	0	36	196	2025	6889
0	0	36	0	784	196	13225
0	1	4	0	81	784	5776
0	4	16	36	0	1681	81
0	0	0	100	784	1024	1369
0	1	16	16	225	3600	1936
0	0	9	64	841	2209	484
0	0	36	144	64	841	0
0	9	0	9	324	64	900
0	1	16	196	49	324	49
0	0	36	64	16	49	2601
0	4	16	16	169	16	484
0	9	16	36	36	169	10201
0	0	0	16	0	36	36
0	1	9	25	0	1024	13924
0	4	16	64	36	1024	15376
0	0	0	9	100	36	11236
0	0	25	144	16	1764	15376
0	4	36	64	576	16	11236
0	4	16	169	784	3136	196
0	9	1	196	9	961	8649
9801	9216	9216	7569	9604	9409	784
123201	123201	121104	116964	118336	83521	70756
451584	447561	442225	451584	425104	425104	308025
1073296	1067089	1071225	1048576	1069156	1030225	990025
2076481	2076481	2073600	2070721	2062096	1990921	2039184
3474496	565504	3467044	3429904	3426201	3297856	3207681
5216656	5207524	5202961	5180176	5125696	4941729	4665600
7182400	7182400	7160976	7107556	7038409	7033104	6697744
9168784	9150625	9162729	9120400	9168784	8988004	8637721
11042329	11022400	11035684	630261025	10916416	10732176	10758400
12638025	12638025	12616704	12574116	12524521	12432676	11771761
13816089	832207104	141376	13712209	13675204	13682601	13417569
14538969	14531344	14516100	14508481	823690000	14364100	340439401
14814801	14814801	718347204	14730244	14676561	14638276	14440000
14707225	827425225	40094224	14638276	745071616	158659216	10923025
14303524	839724484	14258176	14190289	14288400	14122564	14062500
13719616	843263521	41216400	476330625	13667809	13557124	12888100

13032100	13032100	12996025	13024881	12866569	12752041	5475600
12334144	12320100	12292036	12285025	12334144	11923209	12152196
11710084	11689561	11669056	798119001	11600836	11546404	11128896
11182336	11182336	11155600	11142244	11029041	10936249	10517049
10778089	10764961	10771524	861950881	10627600	10458756	10342656
10510564	10504081	10471696	10432900	10361961	10336225	9840769
10387729	10368400	10361961	10361961	10201636	10176100	10291264
10368400	10368400	10342656	10342656	10240000	10125124	10182481
10445824	10445824	10426441	10400625	10445824	10297681	10381284
10595025	10595025	10575504	10582009	10407076	10246401	10413529
10797796	10778089	10758400	10732176	10797796	10400625	10582009
11015761	11015761	10989225	11015761	10837264	11002489	10478169
11235904	11235904	11195716	11182336	11128896	10850436	182601169
11437924	169052004	11424400	11410884	11316496	11397376	11182336
11600836	11600836	11553201	11573604	11485321	11410884	11471769
11723776	11710084	11723776	11703241	11519236	11634921	4923961
11792356	11785489	11751184	11785489	11669056	11716929	11235904
11812969	11799225	11785489	11771761	11806096	11464996	11812969
11799225	11785489	11799225	11730625	11792356	11634921	11002489
11751184	11737476	11723776	11648569	11696400	11424400	11022400
11682724	11662225	11655396	11634921	11471769	11553201	11600836
11607649	11600836	11566801	11573604	11451456	11560000	11512449
11526025	11526025	11478544	11437924	11485321	11526025	11102224
11451456	11431161	11417641	11377129	11424400	11336689	11209104
11383876	11370384	11356900	11289600	11343424	11048976	11282881
11329956	11329956	11289600	11249316	11202409	10949481	11015761
11296321	11282881	11289600	11235904	11195716	11222500	10497600
11276164	11256025	11249316	11222500	11102224	11002489	7001316
11256025	11242609	11222500	11162281	11229201	10929636	10556001
11215801	11195716	11209104	11135569	11075584	10989225	723341025
11029041	11009124	11022400	10956100	10890000	10876804	10647169
10595025	10575504	10575504	10582009	10556001	10582009	173343556
9872164	9853321	9865881	9853321	9853321	9859600	9394225
8779369	8767521	8773444	8773444	8620096	8456464	8369449
7365796	534361	7354944	7300804	7300804	7128900	7102225
5779216	5769604	5779216	5755201	5726449	5726449	5513104
4165681	4157521	4137156	4108729	4048144	3984016	4072324
2683044	961124004	2676496	2660161	2621161	2676496	2634129
1473796	712356100	1466521	1473796	230250276	1382976	1297321
628849	624100	627264	616225	589824	602176	525625
156025	156025	151321	145161	147456	144400	144400
1212658261	12751739404	2946509190	5789411088	3849500082	1333070940	4,785E+09

Srms	Nrms(N+1)3	Nrms(N+1)5	Nrms(N+1)7	Nrms(N+1)9	Nrms(N+1)11	Nrms(N+1)13
136,56	442,84	212,87	298,39	243,31	143,18	271,28

BIT's	SNR	dB	
2	0,10	-1021,83	10,22
4	0,41	-385,57	3,86
6	0,21	-678,90	6,79
8	0,32	-501,67	5,02
10	0,91	-41,11	0,41
12	0,25	-596,19	5,96

Appendix D: Mod-Δ (Noise) Data Analysis

#	I	Q	MD(N+1) 3	MD(N+1) 5	MD(N+1) 7	MD(N+1) 9	MD(N+1)1 1	MD(N+1)1 3
0	3374	3432	120	3	15	22	4	34
1	3366	3422	38	27993	9	3	0	42
2	3362	3410	123	2	27845	23	19	16
3	3360	3397	120	4	22508	29	50	48
4	3356	3386	103	7	31109	7	57	125
5	3348	3376	120	3	9	12	63	22371
6	3321	3367	22371	0	32645	4	46	6
7	3256	3362	5	0	14	31117	0	15
8	3140	3360	84	4	4	24	61	65
9	2963	3357	59	2	1	8	24	69
10	2716	3351	16475	7	3	26	14	59
11	2404	3331	108	5	7	24	6	80
12	2040	3276	90	28422	14	4	3	84
13	1638	3175	61	7	11	29	6	89
14	1215	3014	60	16384	5	12	11	61
15	793	2784	114	1	15	1	33	64
16	395	2487	31	3	31111	21	57	108
17	48	2135	1	1699	10	6	1	126
18	0	1742	51	7	12	22	40	124
19	0	1322	14	3	10	28	52	9
20	0	897	81	518	14	10	48	37
21	0	491	115	7	13	28	33	44
22	0	129	33	3	7	10	23	22
23	0	0	33	6	2	14	1	0
24	0	0	105	5	6	29	19	30
25	0	0	113	6	9	7	51	7
26	0	0	16	6	1	18	12	51
27	0	0	11	4	8	22	9	22
28	0	0	88	2	4	9	37	101
29	0	0	25	4	0	1	44	6
30	0	0	52	2	1	8	22	118
31	0	0	27	6	7	20	0	124
32	0	0	42	5	1	16	30	106
33	0	0	100	7	3	1	7	124
34	0	0	67	2	3	23	51	106
35	0	0	74	6	12	1	22	14
36	0	0	65	1	9	19	37	93

37	0	0	86	1	5	19	6	71
38	0	0	35	0	31116	12	54	114
39	0	0	105	4	6	9	60	22
40	0	0	64	0	0	5	42	9
41	0	0	66	1	14	12	60	1
42	0	0	35	7	7	22	42	40
43	0	0	40	1	3	0	14	52
44	0	0	71	3	6	30	29	112
45	0	0	33	3	5	7	7	97
46	0	0	88	4	6	19	50	87
47	0	0	122	1	6	22	22	1
48	0	0	57	5	12	5	9	83
49	0	0	66	516	10	6	1	115
50	0	0	26	6	12	22	40	76
51	0	0	3	0	10	28	52	9
52	0	0	33	6	14	10	48	37
53	0	0	115	6	13	28	33	44
54	0	0	33	3	7	10	23	22
55	0	0	120	6	2	14	1	0
56	0	0	105	5	6	29	19	30
57	0	0	113	6	9	7	51	30
58	0	0	16	6	1	2	60	51
59	112	0	11	4	8	26	45	22
60	356	0	88	2	8	4	0	85
61	667	0	25	4	11	13	56	106
62	1036	0	36	2	13	4	50	17
63	1436	0	127	16642	3	19	63	8
64	1855	61	69	0	32640	19	1	6
65	2275	289	112	3	6	16	54	59
66	2671	583	95	6	2	9	5	77
67	3026	940	9	5	14	26	15	125
68	3321	1334	36	4	1938	19	5	47
69	3552	1749	69	0	5	13846	9	22368
70	3715	2172	117	2	16655	16	26	16466
71	3812	2576	6002	5	26602	32660	32519	25
72	3851	2942	32	0	11	3	40	22253
73	3838	3253	69	4	5820	20	4	12
74	3784	3501	7	3	6	4095	35	38
75	3705	3682	51	6140	12	28	42	124
76	3612	3794	69	30361	2	16793	20248	6009
77	3515	3847	105	3	0	7	16423	31613
78	3424	3846	81	30357	6	4	16387	18
79	3345	3801	22	28421	7	10	42	97

80	**3284**	**3727**	**116**	**0**	**0**	**16**	**52**	**100**
81	**3243**	**3636**	**34**	**6132**	**5**	**27**	**22**	**71**
82	**3221**	**3539**	**43**	**7**	**1**	**9**	**59**	**119**
83	**3219**	**3445**	**22391**	**4**	**22525**	**28**	**20**	**5822**
84	**3232**	**3363**	**204**	**1**	**18366**	**0**	**38**	**56**
85	**3254**	**3297**	**8**	**4**	**3**	**17**	**54**	**204**
86	**3285**	**3251**	**52**	**6**	**26364**	**13825**	**14**	**76**
87	**3319**	**3225**	**24**	**6**	**16393**	**6**	**25**	**85**
88	**3352**	**3218**	**6015**	**3**	**14**	**18**	**8**	**21**
89	**3381**	**3228**	**70**	**3**	**14**	**18932**	**0**	**31**
90	**3405**	**3248**	**7**	**5**	**14**	**16785**	**23339**	**104**
91	**3423**	**3276**	**22371**	**4**	**7**	**0**	**51**	**99**
92	**3434**	**3310**	**19725**	**7**	**14**	**23**	**19**	**68**
93	**3438**	**3344**	**102**	**1**	**14**	**12**	**16407**	**112**
94	**3435**	**3374**	**22323**	**513**	**12**	**13**	**59**	**100**
95	**3429**	**3400**	**5**	**32736**	**12**	**15**	**27**	**103**
96	**3419**	**3419**	**24**	**3**	**12**	**31**	**557**	**79**
97	**3407**	**3432**	**79**	**26330**	**2**	**4**	**33**	**87**
98	**3395**	**3437**	**40**	**24839**	**25702**	**14**	**42**	**57**
99	**3383**	**3437**	**37**	**1**	**26595**	**14**	**3**	**81**
100	**3373**	**3432**	**16**	**0**	**6**	**26**	**12**	**20**
101	**3367**	**3422**	**25**	**4**	**12**	**2036**	**39**	**21876**
102	**3361**	**3410**	**22394**	**7**	**31117**	**11**	**21**	**25**
103	**3359**	**3398**	**22**	**1**	**5**	**5**	**24**	**55**
104	**3356**	**3386**	**103**	**7**	**28620**	**26**	**63**	**40**
105	**3349**	**3375**	**99**	**2**	**3**	**1989**	**53**	**29**
106	**3321**	**3368**	**66**	**6**	**0**	**14**	**16674**	**61**
107	**3256**	**3362**	**68**	**5**	**11**	**4**	**18**	**45**
108	**3140**	**3359**	**92**	**0**	**10**	**16**	**47**	**40**
109	**2962**	**3357**	**26**	**4**	**7**	**25**	**34**	**37**
110	**2715**	**3351**	**5936**	**4**	**1**	**23**	**22**	**105**
111	**2404**	**3331**	**16446**	**26613**	**10**	**15**	**15**	**28**
112	**2041**	**3276**	**75**	**28671**	**5**	**13**	**49**	**55**
113	**1640**	**3175**	**93**	**7**	**2**	**4**	**63**	**108**
114	**1214**	**3013**	**126**	**2**	**1674**	**15**	**1**	**52**
115	**793**	**2783**	**124**	**1**	**32131**	**7**	**63**	**71**
116	**395**	**2487**	**9**	**16646**	**6073**	**25**	**31**	**62**
117	**47**	**2136**	**49**	**5**	**32524**	**28**	**33**	**55**
118	**0**	**1743**	**58**	**4**	**7**	**10**	**23**	**69**
119	**0**	**1322**	**44**	**1**	**2**	**14**	**1**	**0**
120	**0**	**897**	**24**	**4**	**6**	**29**	**19**	**30**
121	**0**	**491**	**113**	**6**	**9**	**7**	**51**	**7**
122	**0**	**128**	**16**	**6**	**1**	**18**	**12**	**51**

123	0	0	11	4	8	22	9	22
124	0	0	88	2	4	9	37	101
125	0	0	25	4	0	1	44	6
126	0	0	52	2	1	8	22	118
127	0	0	27	6	7	20	0	124
128	0	0	42	5	1	16	30	106
129	0	0	100	7	3	1	7	124
130	0	0	67	2	3	23	51	106
131	0	0	74	6	12	1	22	14
132	0	0	65	1	9	19	37	93
133	0	0	86	1	5	19	6	71
134	0	0	35	0	31116	12	54	114
135	0	0	105	4	6	9	60	22
136	0	0	64	0	0	5	42	9
137	0	0	66	1	14	14	62	1
138	2	0	35	7	9	22	42	40
139	0	0	40	1	3	0	14	54
140	0	0	71	3	6	30	29	112
141	0	0	35	5	5	8	8	97
142	1	0	88	4	7	20	51	87
143	1	2	122	1	7	22	22	2
144	0	0	57	5	12	8	12	84
145	3	0	67	517	13	6	1	115
146	0	0	27	7	12	22	40	79
147	0	0	3	0	10	28	52	9
148	0	1	36	1	14	10	48	37
149	0	0	115	6	13	28	33	44
150	0	2	33	3	7	10	23	22
151	0	1	120	6	2	14	1	0
152	0	0	105	5	6	29	19	30
153	0	0	113	6	9	7	51	30
154	0	0	16	6	1	18	12	51
155	0	0	11	4	8	22	9	22
156	0	0	88	2	4	9	37	101
157	0	0	25	4	0	1	44	6
158	0	0	52	2	1	8	22	118
159	0	0	27	6	7	20	0	124
160	0	0	42	5	1	16	30	106
161	0	0	100	7	3	1	7	124
162	0	0	67	2	3	23	51	106
163	0	0	74	6	12	1	22	14
164	0	0	65	1	9	19	37	93
165	0	0	86	1	5	19	6	71

166	0	0	35	0	31116	12	54	114
167	0	0	105	4	6	9	60	22
168	0	0	64	0	0	5	42	9
169	0	0	66	1	14	12	60	1
170	0	0	35	7	14	22	42	40
171	0	0	40	1	3	0	14	52
172	0	0	71	3	6	30	29	112
173	0	0	33	3	5	7	7	97
174	0	0	88	4	6	19	50	87
175	0	0	122	1	6	22	22	1
176	0	0	57	5	12	5	9	83
177	0	0	66	516	10	6	1	115
178	0	0	26	6	12	22	40	76
179	0	0	3	0	10	28	52	9
180	0	0	33	6	14	10	48	37
181	0	0	115	7	13	28	33	44
182	0	0	33	3	7	10	23	22
183	0	0	120	6	2	14	1	0
184	0	0	105	5	6	29	19	30
185	0	0	113	6	9	7	51	7
186	0	0	16	6	1	18	12	51
187	0	0	11	4	8	22	9	22
188	0	0	88	2	4	9	37	101
189	0	0	25	4	0	1	44	6
190	0	0	52	2	1	8	22	118
191	0	0	27	6	7	20	0	124
192	0	0	42	5	1	16	30	106
193	0	0	100	7	3	1	30	124
194	0	0	67	2	3	23	51	106
195	0	0	74	6	12	1	22	14
196	0	0	65	1	9	22	8	93
197	99	0	86	1	8	18	37	71
198	351	0	35	0	16651	12	22	85
199	672	0	105	4	6	21	8	117
200	1036	0	35	3	12	6	11	41
201	1441	0	33	0	15	20	4	13
202	1864	50	67	7	15	2	22	73
203	2284	280	52	5	15	24	6	124
204	2680	587	104	4	14	18	49	92
205	3028	941	105	3	9	25	2	89
206	3323	1337	68	0	1	22	21	43
207	3555	1758	114	32513	9	27	27	124
208	3717	2181	13	1	1	18890	46	54

209	3813	2585	8189	7	18911	16399	10	22264
210	3849	2946	125	1	5	29073	35	49
211	3835	3255	584	6141	5	2	58	530
212	3782	3504	20230	32715	4	2	40	32
213	3704	3682	124	0	9989	32662	59	114
214	3610	3795	6012	6	6097	2	15	5950
215	3512	3845	103	4	18346	12	31	26
216	3422	3842	24881	5	4	13	35	86
217	3344	3798	20235	0	9	16858	6	101
218	3283	3725	6008	6	4	2044	54	67
219	3242	3634	105	2	2	13	16416	105
220	3223	3536	104	2	11	29	57	15
221	3220	3443	108	7	4	1	12	29
222	3232	3362	94	20444	1	31	13	63
223	3255	3296	6002	30637	14	10	22	36
224	3286	3250	62	1	7	7	21	15
225	3319	3226	4	30639	10	25	31	47
226	3352	3219	122	26609	28603	13	41	121
227	3382	3228	32	4	26594	15	36	6004
228	3406	3248	56	0	6135	19	5	106
229	3424	3278	110	1	5	18909	48	30244
230	3434	3311	22393	6	518	1977	35	58
231	3437	3343	55	2	3	20	39	16421
232	3435	3375	32	0	30715	9	14	77
233	3428	3400	44	3	18802	6134	22	70
234	3418	3420	16	4	1	5	16441	49
235	3407	3432	16403	4	2	18883	17	56
236	3395	3437	43	7	9	18006	21	23
237	3384	3437	123	32757	13	21	53	15
238	3374	3430	5863	30627	4	25	16408	75
239	3366	3421	61	4	12	23	55	68
240	3361	3410	113	5	13	3	39	15
241	3358	3398	112	30410	8	1	28	7
242	3355	3387	64	4	7	11	61	25
243	3349	3376	1636	30697	29967	21	45	124
244	3321	3367	16447	25580	18311	1	39	106
245	3255	3362	14	1	4	2	39	14
246	3142	3359	54	0	27901	29	42	93
247	2963	3356	113	7	15733	1992	27	71
248	2714	3352	8176	4	0	1	55	114
249	2404	3331	16695	4	13	0	44	22
250	2041	3275	35	1	10	24	50	9
251	1638	3176	37	16390	16398	20	7	1

252	**1214**	**3015**	**60**	**6**	**2**	**2**	**62**	**40**
253	**793**	**2782**	**18**	**5**	**9**	**12**	**55**	**52**
254	**395**	**2486**	**26**	**0**	**16652**	**52**	**5**	**112**