Preprocessing for Extracting Signal Buried in Noise Using LabVIEW

A Thesis report

submitted towards the partial fulfillment of the

requirements of the degree of

Master of Engineering

in

Electronics Instrumentation and Control Engineering

submitted by

Nitish Sharma

Roll No-800851013

Under the supervision of

Dr. Yaduvir Singh
Associate Professor, EIED
and
Dr. Hardeep Singh
Assistant Professor, ECED

DEPARTMENT OF ELECTRICAL AND INSTRUMENTATION ENGINEERING

THAPAR UNIVERSITY

PATIALA- 147004

JULY 2010

DECLARATION

I hereby declare that the report entitled ***"Preprocessing for Extracting Signal Buried in Noise Using LabVIEW"*** *is an authentic record of my own work carried out as requirements for the award of degree of M.E. (Electronic Instrumentation & Control) at Thapar University, Patiala, under the guidance of Dr. Yaduvir Singh (Associate Professor, EIED) and Dr. Hardeep Singh (Assistant Professor, ECED) during January to July 2010.*

Date: 13/07/10

Nitish Sharma

Nitish Sharma

Roll No.- 800851013

It is certified that the above statement made by the candidate is correct to best of my knowledge and belief.

13/07/2010

Dr. Yaduvir Singh
Associate Professor, EIED,
(Supervisor)
Thapar University, Patiala

H.Singh

Dr. Hardeep Singh
Assistant Professor, ECED,
(Co-Supervisor)
Thapar University, Patiala

S. Ghosh 13/7/10

Dr. Smarajit Ghosh
Professor & Head, EIED,
Thapar University, Patiala

R.K. Sharma 14.7.10

Dr. R.K. Sharma
Dean of Academic Affairs
Thapar University, Patiala

i

ACKNOWLEDGEMENT

Words are often too less to reveals one's deep regards. An understanding of the work like this is never the outcome of the efforts of a single person. I take this opportunity to express my profound sense of gratitude and respect to all those who helped me through the duration of this thesis.

First of all I would like to thank the Supreme Power, one who has always guided me to work on the right path of the life. Without His grace this would never come to be today's reality.

This work would not have been possible without the encouragement and able guidance of my supervisor, ***Dr. Yaduvir Singh****, Associate Professor, EIED, Thapar University, Patiala and Co-supervisor,* ***Dr. Hardeep Singh****, Assistant Professor, ECED, Thapar University, Patiala. Most of the novel ideas and solutions found in this thesis are the result of our numerous stimulating discussions. Their feedback and editorial comments were also invaluable for the writing of this thesis.*

I would like to express my deep sense of gratitude towards ***Dr. Smarajit Ghosh****, Professor and Head, EIED, Thapar University, Patiala who has been a constant source of inspiration for me throughout this work.*

I would also like thank all the faculty members of the department and my friends who directly or indirectly helped me in completion of my thesis. No words of thanks are enough for my dear parents and sister whose support and care makes me stay on earth. Thanks to be with me.

Date: 13/07/10
Place: Thapar University, Patiala

Nitish Sharma
Nitish Sharma

ii

ABSTRACT

Digital signals are used everywhere in the world around us due to its superior fidelity, noise reduction, and signal processing flexibility to remove noise and extract useful information. This unwanted electrical or electromechanical energy that distort the signal quality is known as noise, it can block, distort change or interfere with the meaning of a message in both human and electronic communication. Engineers are constantly striving to develop better ways to deal with noise disturbances as it leads to loss of useful information. The traditional method has been used to minimize the signal bandwidth to the greatest possible extent. However reducing the bandwidth limits the maximum speed of the data than can be delivered. Another, more recently developed scheme for minimizing the effect of noise is called digital signal processing.

Digital filters are very much more versatile in their ability to process signals in a variety of ways and can handle low frequency signals accurately; this includes the ability of some types of digital filter to adapt to changes in the characteristics of the signal. Fast DSP processors can handle complex combinations of filters in parallel or cascade (series), making the hardware requirements relatively simple and compact in comparison with the equivalent analog circuitry. In order to extract useful information from the noisy signals, raw signal has to be processed. By analyzing and processing the digital data, the useful information is obtained from the noise and present it in a form more comprehensible than the raw data. The signal processing can be roughly divided into two stages by functionality: preprocessing and feature extraction. The preprocessing stage removes or suppresses noise from the raw signal and the feature extraction stage extracts information from the signal. In this thesis the extraction of a signal i.e. preprocessing is done by using LabVIEW which has extensive set of signal processing VI's that simplify the development of digital processing system for improved noise reduction and efficient extraction of desired signal.

ORGANIZATION OF THESIS

Chapter-1 It includes the introduction about the data analysis for digital signal processing of analog signal.

Chapter-2 This chapter describes the various types of noises and related terms along with noise measurement and calculation.

Chapter-3 LabVIEW basics and its programming has been discussed in this chapter.

Chapter-4 This chapter describes the digital filters and its various types along with their design and implementations chapter.

Chapter-5 The result and discussion has been described in this chapter.

Chapter-6 Thesis has been concluded with future scope in this chapter.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

NOMENCLATURE

MOS	Metal Oxide Semiconductor
DSP	Digital Signal Processing
A/D	Analog to digital convertor
IF	Input frequency
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
f_s	Sampling frequency
f_n	Nyquist frequency
K_B	Boltzmann constant
q	Elementry charge
SNR	Signal to noise ratio
h	Plank's constant
AWGN	Additive White Gaussian noise
MOP	Multi objective optimization problem
N	Average Number of photons selected
PDF	Probability density function
$S_x(w)$	Spectral density
CIMSF	Closet integer multiple of the sampling frequencynt
x(t)	Signal
w	White random

$\Psi(f)$	Fourier transform of F(t)
R	Resistance
C	Capacitor
K	Process dependent constant
NIHL	Noise induced hearing loss
RJ	Random Jitter
DR	Dynamic Range
L	Attenuation
LCCD	linear constant-coefficient different equation
D(w)	Group delay
DAQ	Data acquisition
AF	Alias frequency
NF	Noise figure
W	Channel width

LITERATURE SURVEY

J.B. Johnson (1928). He had discussed about statistical fluctuation of electric charge exists in all conductors, producing random variation of potential between the ends of the conductor. The effect of these fluctuations has been measured by a vacuum tube amplifier and thermocouple, and can be expressed by the formula:

$$I^2 = (2kT) \int_0^\infty R(w) |Y(w)^2| dw$$

I is the observed current in the thermocouple, k is Boltzmann's gas constant, T is the absolute temperature of the conductor, R(*w*) is the real component of impedance of the conductor, Ye (*w*) is the transfer impedance of the amplifier, and *w/2Π=f* represents frequency. The value of Boltzmann's gas constant obtained from the mesurements lie near the accepted value of this constant. The technical aspects of the disturbances were discussed. In an amplifier having a range of 5000 cycles and the input resistance R the power equivalent of the effects is $\frac{\bar{V}^2}{R} = 0.8 * 10^{-16}$ watt, with corresponding power for other ranges of frequency. The least contribution of tube noise is equivalent to that of a resistance R_c=1.5 X 10^5 i_p/ur, where i_p is the space current in milliamperes and u is the effective application of tube [1].

P.K. Nanda Et. al. (1991). They purposed about estimation of unknown frequency, extraction of narrowband signals buried under noise and periodic interference are accomplished by employing the existing techniques. However, this paper proposed an artificial neural net based scheme together with pattern classification algorithm for narrowband signal extraction, A three layer feedforward net is trained with three different algorithms namely backpropagation, Cauchy's algorithm with Boltzmann's probability distribution feature and the combined backpropagation-cauchy's algorithm. Constrained tangent hyperbolic function is used to activate individual neuron. Computer simulation is carried out with inadequate data to reinforce the idea of net's generalization capability. The robustness of the proposed scheme is claimed with the results obtained by making 25% links faulty between the layers. Performance comparison of the three algorithms is made and the superiority of the combined backpropagation-Cauchy's algorithm is

established over the other two algorithms. Simulation results for a wide variety of cases are presented for better appraisal [2].

Gary E. Birch Et. al. (1994). In this paper, Initial investigations of Birch's outlier processing method (OPM) have demonstrated an ability to extract a special class of finite duration signals from colored noise processes. This special class of signals are low SNR signals whose shapes and latencies can vary dramatically from trial to trial. Such signals, termed highly variable events (HVE) in this paper, are commonly found in physiological signal analysis applications. This work identifies that the OPM produces suboptimal HVE estimates due to its use of time-invariant influence functions and demonstrates that the addition of time-varying influence functions (TVIF's) produce improved estimates. Simulation experiments with signals in white and colored noise processes were used to demonstrate the modified OPM algorithm's superior performance compared to the performance of the original algorithm and to the performance of a time-invariant minimum mean-square-error (MMSE) filter for linear and stationary process. The simulation results indicated that the OPM algorithm with TVIF's can extract HVE's from a linear and stationary process for SNR levels above -2.5 dB and can work effectively as low as -10.0 dB in certain situations [3].

Giacinto Gelli Et. al. (1997). Their paper deals with signal extraction performed by processing data received by an array of sensors. The proposed method was blind, i.e., it does not require any a priori knowledge of directional information associated with the signals of interest (SOI's). Such information was obtained directly from the received data by exploiting the higher-order cyclostationarity (HOCS) properties exhibited by most communication signals. The proposed method was inherently tolerant to both possibly non-stationary Gaussian disturbances as well as non-Gaussian interferences not exhibiting the same HOCS properties presented by the SOI's. Simulation results confirmed the effectiveness of the method when it operates in severely degraded disturbance environments [4].

N. Hemel Et. al. (1999). In this paper, they developed a new computationally efficient method, based on multirate techniques, for heart rate variability (HRV) signal extraction from the R-R intervals is presented [5].

Jiann-An Tsai Et. al. (2001). In this paper, they proposed a new signal extraction method that combines MMSE beam forming with successive interference cancelation (SIC) for an overloaded antenna array system. By an overloaded array system, its mean that the number of desired signals exceeds the number of distinct antenna elements, often by a significant amount. They evaluated the performance of the proposed signal extraction method through the use of the bit error rate (BER) simulation. The simulation results showed that the new signal extraction method can increased the number of users that can be supported in an airborne communication environment. They also evaluated the effects of directional antenna patterns on the performance of the proposed signal extraction method. Performance results for an 8 element antenna array show that the proposed signal extraction method with a 60^0 half-power beam width (HPBW) directional pattern can support up to 16 users (twice as many as number of antenna element) and a 30^0 HPBW directional pattern can support up to 24 users three times as many as number of antenna element [6].

Shan Ouyang Et. al. (2003). In this paper, they begin with reviewing the concept of cyclostationarity and then illustrate that a conventional blind fresh filter is not suitable to apply in project because the computational complexity was very high. A reduced-rank method was developed which is based on MSWF. Then constructed a test model to verify the algorithm performance which implements in DSP. The Matlab simulation and the statistical outcome of DSP indicate that the method could extract
spectral overlapping interference signal [7].

J.M. Graybeal Et. al. (2004). In this paper, they discussed about pulse oximetry utilizing spectrophotometric principles and normalized absorption of red and infrared light, provides vital information concerning patients' arterial oxygen saturation, pulse rate and perfusion level. Conventional pulse oximeters, incorporating conventional filters, are

hampered by artifact interference from motion, electrical and ambient light and other conditions producing weak signals. They introduced mathematically and physiologically based designs along with adaptive filtering and what it calls DST™ (Discrete Saturation Transform™) as a solution to monitoring patients even during times of severe and unpredictable noise interference. This combined with 4 other alternative calculations, revolutionized pulse oximetry performance. This new technology is called Signal Extraction Pulse Oximetry® or Masimo SET® pulse oximetry. Sensitivity and specificity of signal extraction technology, was first tested extensively in the lab dddon volunteers under conditions designed to simulate varying physiology, including controlled desaturations, combined with severe patient motion, and low perfusion conditions. Conventional pulse oximeters demonstrated very low sensitivity and specificity while pulse oximeters with SET showed sensitivity and specificity of over 95% under the same conditions. Clinical testing was then performed on an extensive variety of patients in the hospital environment demonstrating similar performance, validating the significant advance resulting from the use of SET. False alarms due to motion artifact and low perfusion have been reduced from up to 90% to less than 5% [8].

Shun-ichi Amari Et. al. (2004). In this paper, they study on the problem of the blind simultaneous extraction of specific groups of independent components from a linear mixture. This paper first presents a general overview and unification of several information theoretic criteria for the extraction of a single independent component. Their contribution fills the theoretical gap that exists between extraction and separation by presenting tools that extend these criteria to allow the simultaneous blind extraction of subsets with an arbitrary number of independent components. In addition, they analyze a family of learning algorithms based on Stiefel manifolds and the natural gradient ascent, present the nonlinear optimal activations (score) functions, and provide new or extended local stability conditions. Finally, they illustrate the performance and features of the proposed approach by computer-simulation experiments [9].

Hagit Messer Et. al. (2005). They presented a simplified model for a periodic signal that propagates through a random medium, the received signal is mixed with a background

noise, and in addition, each period is randomly time shifted and attenuated. In this correspondence, they introduced two methods for retrieving the magnitude spectrum of the nominally periodic waveform from repeated noisy measurements and estimating some of the parameters that characterize the random medium. The first method is based on averaging the biperiodograms of the noisy data. They showed that the reconstructed magnitude spectrum is an unbiased and consistent estimator if the background noise is white with a symmetric pdf. The second method is based on averaging the periodograms of the noisy data. In this method, it is possible to reconstruct the magnitude spectrum only if the magnitude of the background noise is either known or can be estimated from an independent measurements. Both methods are analyzed, and their performance was demonstrated via Monte Carlo simulations [10].

Muhammad I. Ibrahimy Et. al. (2006). They presented paper to model the developed algorithms on FECG signal extraction using VHDL (Very High Speed Integrated Circuit Hardware Description Language) for FPGA (Field Programmable Gate Array) implementation from the abdominal ECG signal using Neural Network approach was to provide efficient and effective ways of separating FECG signal and its nature. FECG (Fetal ECG) signal contains potentially precise information that could assist clinicians in making more appropriate and timely decisions during pregnancy and labour. The extraction and detection of the FECG signal from composite abdominal signals with powerful and advance methodologies is becoming a very important requirement in fetal monitoring [11].

P. C. Ching Et. al. (2006). In this paper, the application of dyadic wavelet decomposition in the context of time delay estimation was investigated. They consider a model in which the source signal is deterministic and the received sensor outputs are corrupted by additive noises. Wavelet denoising was exploited to provide an effective solution for the problem. Denoising is first applied to preprocess the received signals from two spatially separated sensors with an attempt to remove the contamination, and the peak of their cross correlation function is then located from which the time delay between the two signals can be derived. A novel wavelet shrinkage/thresholding technique for denoising is

introduced, and the performance of the algorithm is analyzed rigorously. It was proved that the proposed method achieves global convergence with a high probability. Simulation results also corroborate that the technique is efficient and performs significantly better than both the generalized cross correlator (GCC) and the direct cross correlator [12].

Chih-Kuo Liang Et. al. (2006). They presented paper on research to design an EEG BCI (Brain Computer Interface) system and develop techniques for helping the serious disabled with spine/central nerve injury, motor neuron disease or without upper limb/foot. In their research, the brain wave amplifier was used is the NuAmps amplifier and the acquisitive/analytic software for EEG is named "Scan 4.3". Both of them were developed by NeuroScan Company. The other software they used to implement the human-machine interface and on-line analysis functions is LabVIEW 8.2.1 version developed by National Instruments. They programmed TCP/IP communication protocol to connect Scan 4.3 with LabVIEW. In their experiments, They focused on VEP (Visual Evoked Potential) to design the EEG BCI system. There are four directional arrows (up, down, left, right) shown on the monitor randomly for stimulating the subject. Regarding data analysis, they used ICA (Independent component Analysis) to filter the artifact of EOG/EMG, and calculate the average value after the summation of all trials of EEG signals. Therefore, they found some special phenomena and regarded them as VEP EEG features[13].

Kianoush Nazarpour Et al. (2007). They proposed novel iterative blind signal extraction (BSE) scheme for the removal of the eye-blink artifact from electroencephalogram (EEG) signals. In this method, in order to remove the artifact, the signal extraction algorithm is provided with a priori information, i.e., an estimation of the column of the mixing matrix corresponding to the eyeblink source. Then priori knowledge, namely the vector corresponding to the spatial distribution of the eye-blink factor, is identified by using the method of parallel factor analysis (PARAFAC). Hence, we call the BSE approach, semiblind signal extraction (SBSE). The results demonstrate that the proposed algorithm effectively identifies to removes the eye-blink artifact from raw EEG measurements [14].

U Fai Chan Et. al. (2007). Their paper introduced general front-end amplifier for various bioelectric signals based on Field Programmable Analogy Array (FPAA) Technology. Employing FPAA technology, the implemented amplifier can be adapted for various bioelectric signals without alternating the circuitry while its compact size (core parts < 2 cm2) provides an alternative solution for miniaturized Telemedicine system and Wearable Devices. The proposed design implementation has demonstrated, through successfully ECG and EMG signal extractions, a quick way to miniaturize analog biomedical circuit in a convenient and cost effective way [15].

Philip Langley Et. al. (2007). Their paper assessed and compared three algorithms for extracting the trial signal from the 12-lead ECG. The 12-lead ECGs of 30 patients in atrial fibrillation were analyzed. The atrial activity was extracted by three algorithms, Spatiotemporal QRST cancellation (STC), principal component analysis (PCA), and independent component analysis (ICA). The amplitude and frequency characteristics of the extracted atrial signals were compared between algorithms and against reference data. Mean (standard deviation) amplitude of QRST segments of V1 was 0.99 (0.54) mV, compared to 0.18 (0.11) mV (STC), 0.19 (0.13) mV (PCA), and 0.29 (0.22) mV (ICA). Hence, for all algorithms there were significant reductions in the amplitude of the ventricular activity compared with that in V1. Reference atrial signal amplitude in V1 was 0.18 (0.11) mV, compared to 0.17 (0.10) mV (STC), 0.12 (0.09) mV (PCA), and 0.18 (0.13) mV (ICA) in the extracted atrial signals. PCA tended to attenuate the atrial signal in these segments. There were no significant differences for any of the algorithms when comparing the amplitude of the reference atrial signal with that of the extracted atrial signals in segments in which ventricular activity had been removed. There were no significant differences between algorithms in the frequency characteristics of the extracted atrial signals. There were discrepancies in amplitude and frequency characteristics of the atrial signal in only a few cases resulting from notable residual ventricular activity for PCA and ICA algorithms [16].

QI Ai-ling Et. al. (2007). They had discussed about Interference noising originating from the ultrasonic testing defect signal seriously influences the accuracy of the signal

extraction and defect location. Sparse signal representations are the most recent technique in the signal processing. This technique is utilized to extract casting ultrasonic flaw signals in this paper. But its calculation is huge. A new improved matching pursuit algorithm is proposed. Artificial fish swarm algorithm is a stochastic global optimization technique proposed lately. A hybrid artificial fish swarm optimization algorithm based on mutation operator and simulated annealing are employed to search the best atomic, it can greatly reduce complexity of sparse representations. Experimental results to detect ultrasonic flaw echoes contaminated by white Gaussian additive noise or correlated noise were presented in the paper. Compared with the wavelet transform, the results showed that the signal quality and performance parameters are improved obviously [17].

Jorge E. Monzon Et. al. (2007). In this paper, they presented a method for blind source separation (BSS) of electrocardiographic signals disturbed by noise, using the Lyapunov stability conditions for linear time invariant systems. They used records from the MIT Arrhythmia Database and in-vivo records from cardiac voluntary patients to test their signal extraction method. They quantitatively assessed the performance of the separation procedure. The method presented here compares favorably to other BSS algorithms [18].

Toshihisa Tanaka Et. al. (2007). They proposed signal extraction method from multi-channel EEG signals and apply to extract Steady State Visually Evoked Potential (SSVEP) signal. SSVEP is a response to visual stimuli presented in the form of flushing patterns. By using several flushing patterns with different frequency, brain machine (computer) interface (BMI/BCI) can be realized. Therefore it was important to extract SSVEP signals from multichannel EEG signals. At first, they estimate the power of the objective signal in each electrode. Estimation of the power was helpful in not only extraction of the signal but also drawing a distribution map of the signal, finding electrodes which have large SNR, and ranking electrodes in sort of information with respect to the power of the signal. Experimental results show that the proposed method 1) estimates more accurate power than existing methods, 2) estimates the global signal which has larger SNR than existing methods, and 3) allows us to draw a distribution map of the signal, and it conforms the biological theory [19].

Rahman Jamal Et. al. (2007). In this paper,They discussed that importance of digital filters is well established. Digital filters, and more generally digital signal processing algorithms, are classified as discrete-time systems. They are commonly implemented on a general purpose computer or on a dedicated digital signal processing (DSP) chip. Due to their well-known advantages, digital filters are often replacing classical analog filters. In this application note, they introduce a new digital filter design and analysis tool implemented in LabVIEW with which developers can graphically design classical IIR and FIR filters, interactively review filter responses, and save filter coefficients. In addition, real-world filter testing can be performed within the digital filter design application using a plug-in data acquisition board [20].

E.Vassalos Et. al. (2008). The purpose of this paper is to present the LabVIEW to CCS Link which is a developed from scratch toolkit. It provides the user with the opportunity to design and implement in a fast Communication and easy way, interfaces with LabVIEW 7.1 or later, for digital signal and image processing applications on DSPs. The interfaces (VIs) created with LabVIEW to CCS Link can setup and control the subVIs that belong to the Setup category, provide the Code Composer Studio v3.1 or later and exchange data with more than one DSPs. In order to demonstrate the LabVIEW to CCS user with the capability to create new VIs with LabVIEW that Link capabilities, the implementation of two typical applications will control the CCStudio v3.1 Setup and define automatically on signal and image processing are also presented [21].

Jani Even Et. al. (2008). This paper presents a new blind signal extraction method based on mutual information. Conventional blind signal extraction methods minimize the mutual information between the extracted signal and the remaining signals indirectly by using a cost function. The proposed method directly minimizes this mutual information through a gradient descent. The derivation of the gradient exploits recent results on the differential of the mutual information and the implementation is based on kernel based density estimation. Some simulation results show the performance of the proposed approach and underline the improvement obtained by using the proposed method as a post-processing for conventional methods [22].

CHAPTER - 1

INTRODUCTION

Digital signals are gaining importance in the world. Telephone companies use digital signals to represent the human voice. Radio, television, and hi-fi sound systems are all gradually converting to the digital domain because of its superior fidelity, noise reduction, and signal processing flexibility. Data is transmitted from satellites to earth ground stations in digital form. NASA pictures of distant planets and outer space are often processed digitally to remove noise and extract useful information. Economic data, census results, and stock market prices are all available in digital form. Because of the many advantages of digital signal processing, analog signals also are converted to digital form before they are processed with a computer.

1.1 DATA ANALYSIS

The importance of integrating analysis libraries into engineering stations is that the raw data, as shown in Figure 1.1, does not always immediately convey useful information. Often, to transform the signal, remove noise disturbances, correct for data corrupted by faulty equipment, or compensate for environmental effects, such as temperature and humidity.

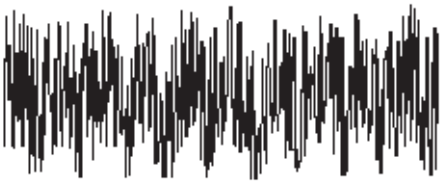

Figure 1.1 Raw Data

By analyzing and processing the digital data, the useful information from the noise and present it in a form more comprehensible than the raw data, as shown in Figure 1.2. Using LabVIEW block diagram programming approach and extensive set of LabVIEW signal processing and measurement VI's simplify the development of analysis applications.

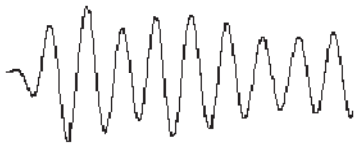

Figure 1.2 Processed Data

1.1.1 Sampling Signal

For the measurement of frequency content of a signal requires digitalization of a continuous signal. In the digital signal processing techniques, it must first convert an analog signal into its digital representation. In practice, the conversion is implemented by using an analog-to-digital (A/D) converter. Consider an analog signal $x(t)$ that is sampled every t seconds. The time interval Δt is the sampling interval or sampling period. Its reciprocal, $1/t$, is the sampling frequency, with units of samples/second. Each of the discrete values of $x(t)$ at $t = 0, \Delta t, 2\Delta t, 3\Delta t$, and so on, is a sample. Thus, $x(0)$, $x(\Delta t)$, $x(2\Delta t)$, …, are all samples. The signal $x(t)$ thus can be represented by the following discrete set of samples. $\{x(0), x(\Delta t), x(2\Delta t), x(3\Delta t), \ldots, x(k\Delta t), \ldots\}$ Figure 1.3 shows an analog signal and its corresponding sampled version. The sampling interval is Δt. The samples are defined at discrete points in time.

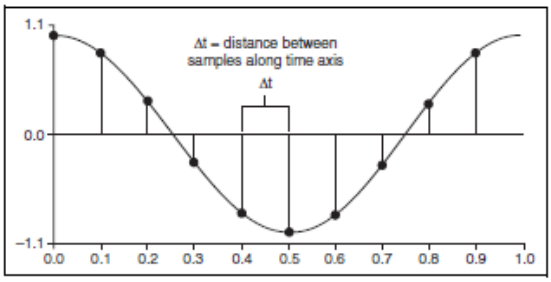

Figure 1.3 Analog Signal and Corresponding Sampled Version

The following notation represents the individual samples.

$$x[i] = x(i\Delta t) \text{ for } i = 0, 1, 2, \ldots n \qquad (1.1)$$

If N samples are obtained from the signal $x(t)$, then it can represent $x(t)$ by the following sequence.

$$X = \{x[0], x[1], x[2], x[3], \ldots, x[N-1]\} \qquad (1.2)$$

The preceding sequence representing $x(t)$ is the digital representation, or the sampled version, of $x(t)$. The sequence $X = \{x[i]\}$ is indexed on the integer variable i and does not contain any information about the sampling rate. So knowing only the values of the

samples contained in *X* gives no information about the sampling frequency. One of the most important parameters of an analog input system is the frequency at which the DAQ device samples an incoming signal. The sampling frequency determines how often an A/D conversion takes place. Sampling a signal too slowly can result in an aliased signal.

1.1.2 Aliasing

An aliased signal provides a poor representation of the analog signal. Aliasing causes a false lower frequency component to appear in the sampled data of a signal. Figure 1.4 shows an adequately sampled signal and an undersampled signal.

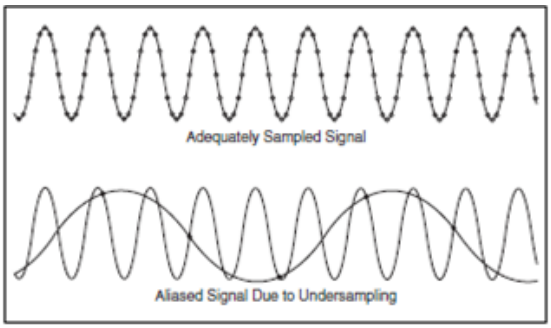

Figure 1.4 Aliasing Effects of an Improper Signal Rate

In Figure 1.4, the undersampled signal appears to have a lower frequency than the actual signal—three cycles instead of ten cycles. Increasing the sampling frequency increases the number of data points acquired in a given time period. Often, a fast sampling frequency provides a better representation of the original signal than a slower sampling rate. For a given sampling frequency, the maximum frequency you can accurately represent without aliasing is the Nyquist frequency. The Nyquist frequency equals one-half the sampling frequency, as shown by the following equation:

$$f_n = \frac{f_s}{2} \qquad (1.3)$$

where f_n is the Nyquist frequency and f_s is the sampling frequency. Signals with frequency components above the Nyquist frequency appear aliased between DC and the Nyquist frequency. In an aliased signal, frequency components actually above the Nyquist frequency appear as frequency components below the Nyquist frequency. For example, a component at frequency $f_n < f_0 < f_s$ appears as the frequency $f_s - f_0$.

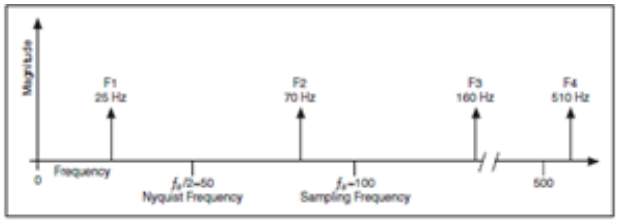

Figure 1.5 Actual Signal Frequency Components

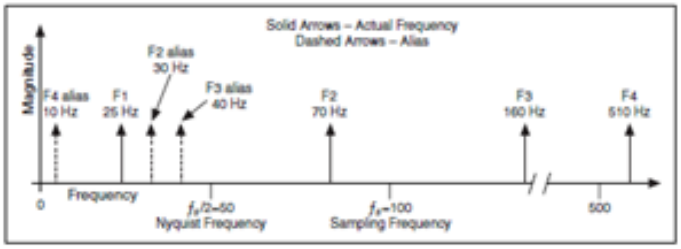

Figure 1.6 Signal Frequency Components and Aliases

Above Figures 1.5 and 1.6 illustrate the aliasing phenomenon. Figure 1.5 shows the frequencies contained in an input signal acquired at a sampling frequency, f_s of 100 Hz. In Figure 1.6, frequencies below the Nyquist frequency of $f_s/2$ = 50 Hz are sampled correctly. For example, F1 appears at the correct frequency. Frequencies above the Nyquist frequency appear as aliases. For example, aliases for F2, F3, and F4 appear at 30 Hz, 40 Hz, and 10 Hz, respectively. The alias frequency equals the absolute value of the difference between the closest integer multiple of the sampling frequency and the input frequency, as shown in the following equation:

$$AF = |CIMF - IF| \quad (1.4)$$

where *AF* is the alias frequency, *CIMSF* is the closest integer multiple of the sampling frequency, and *IF* is the input frequency. For example, computation of the alias frequencies for F2, F3, and F4 from Figure 1.6 with the following equations:

$$\text{Alias F2} = |100 - 70| = 30 \text{ Hz} \quad (1.5)$$

$$\text{Alias F3} = |(2)100 - 160| = 40 \text{ Hz} \quad (1.6)$$

$$\text{Alias F4} = |(5)100 - 510| = 10 \text{ Hz} \quad (1.7)$$

1.1.3 Increasing Sampling Frequency to Avoid Aliasing

According to the Shannon Sampling Theorem, Use a sampling frequency at least twice the maximum frequency component in the sampled signal to avoid aliasing. Figure 1.7 shows the effects of various sampling frequencies. In case A of Figure 1.7, the sampling

frequency f_s equals the frequency f of the sine wave f_s is measured in samples/second. f is measured in cycles/second. Therefore, in case A, one sample per cycle is acquired. The reconstructed waveform appears as an alias at DC. In case B of Figure 1.7, f_s is = 7/4f, or 7 samples/4 cycles. In case B, increasing the sampling rate increases the frequency of the waveform. However, the signal aliases to a frequency less than the original signal—three cycles instead of four. In case C of Figure 1.7, increasing the sampling rate to $f_s = 2f$ results in the digitized waveform having the correct frequency or the same number of cycles as the original signal. In case C, the reconstructed waveform more accurately represents the original sinusoidal wave than case A or case B. By increasing the sampling rate to well above f, for example, $f_s = 10f$ = 10 samples/cycle, you can accurately reproduce the waveform. Case D of Figure 1.7 shows the result of increasing the sampling rate to $f_s = 10f$.

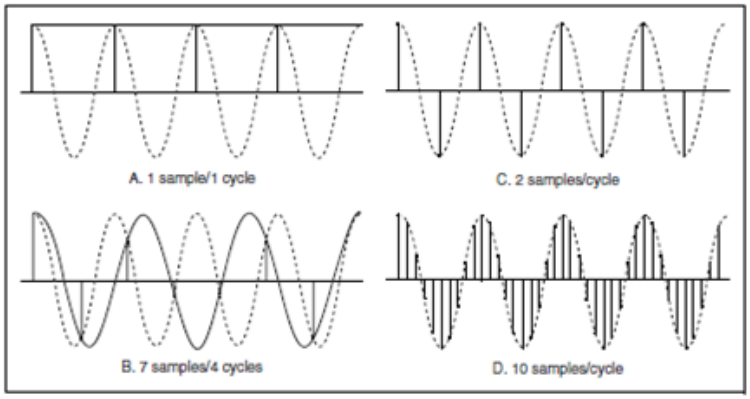

Figure 1.7 Effects of Sampling at Different Rates

1.1.4 Anti-Aliasing Filter

The problem with the digital domain is that it can't distinguish alias frequencies from the frequencies that actually lie between 0 and the Nyquist frequency. Even with a sampling frequency equal to twice the Nyquist frequency, pickup from stray signals, such as signals from power lines or local radio stations, can contain frequencies higher than the Nyquist frequency. Frequency components of stray signals above the Nyquist frequency might alias into the desired frequency range of a test signal and cause erroneous results. Therefore, there is need to remove alias frequencies from an analog signal before the signal reaches the A/D converter. Use an anti-aliasing analog low pass filter before the A/D converter to remove alias frequencies higher than the Nyquist frequency. A low pass

filter allows low frequencies to pass but attenuates high frequencies. By attenuating the frequencies higher than the Nyquist frequency, the anti-aliasing analog low pass filter prevents the sampling of aliasing components. An anti-aliasing analog low pass filter should exhibit a flat pass band frequency response with a good high-frequency alias rejection and a fast roll-off in the transition band. Because the anti-aliasing filter is applied to the analog signal before it is converted to a digital signal, the anti-aliasing filter is an analog filter. Figure 1.8 shows both an ideal anti-alias filter and a practical anti-alias filter. The following information applies to Figure 1.8:

• f1 is the maximum input frequency.

• Frequencies less than f1 are desired frequencies.

• Frequencies greater than f1 are undesired frequencies.

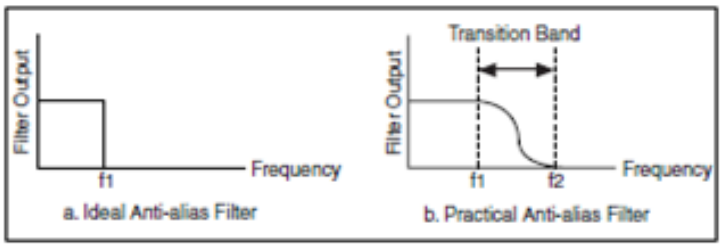

Figure 1.8 Ideal versus Practical Anti-Alias Filter

An ideal anti-alias filter, shown in Figure 1.8, passes all the desired input frequencies and cuts off all the undesired frequencies. However, an ideal anti-alias filter is not physically realizable. Practical anti-alias filters pass all frequencies <f1 and cut off all frequencies >f2. The region between f1 and f2 is the transition band, which contains a gradual attenuation of the input frequencies. Although to pass only signals with frequencies <f1, the signals in the transition band might cause aliasing. Therefore, in practice, use a sampling frequency greater than two times the highest frequency in the transition band. Using a sampling frequency greater than two times the highest frequency in the transition band means f_s might be more than $2f_1$.

CHAPTER - 2

NOISE

2.1 TYPES OF NOISE

In both analog and digital electronics, noise or signal noise is an unwanted random addition to a wanted signal; it is called noise as a generalization of the audible noise heard when listening to a weak radio transmission. Noise is unwanted electrical or electromagnetic energy that degrades the quality of signals and data. Signal noise is heard as acoustic noise if played through a loudspeaker; it manifests as 'snow' on a television or video image. In signal processing or computing it can be considered unwanted data without meaning; that is, data not being used to transmit a signal, but is simply produced as an unwanted by-product of other activities. In Information Theory, however, noise is still considered to be information. In a broader sense, film grain or even advertisements encountered while looking for something else can be considered noise. In biology, noise can describe the variability of a measurement around the mean, for example transcriptional noise describes the variability in gene activity between cells in a population.

Noise can block, distort, change or interfere with the meaning of a message in both human and electronic communication. In a hard-wired circuit such as a telephone-line-based Internet hookup, external noise is picked up from appliances in the vicinity, from electrical transformers, from the atmosphere, and even from outer space. Normally this noise is of little or no consequence. However, during severe thunderstorms, or in locations where many electrical appliances are in use, external noise can affect communications. In an Internet hookup it slows down the data transfer rate, because the system must adjust its speed to match conditions on the line. In a voice telephone conversation, noise rarely sounds like anything other than a faint hissing or rushing.

Noise is a more significant problem in wireless systems than in hard-wired systems. In general, noise originating from outside the system is inversely proportional to the frequency, and directly proportional to the wavelength. At a low frequency such as 300 kHz, atmospheric and electrical noise are much more severe than at a high frequency like 300 megahertz. Noise generated inside wireless receivers, known as internal noise,

is less dependent on frequency. Engineers are more concerned about internal noise at high frequencies than at low frequencies, because the less external noise there is, the more significant the internal noise becomes.

Nowadyas, engineers are constantly striving to develop better ways to deal with noise. The traditional method has been to minimize the signal bandwidth to the greatest possible extent. The less spectrum space a signal occupies, the less noise is passed through the receiving circuitry. However, reducing the bandwidth limits the maximum speed of the data that can be delivered. Another, more recently developed scheme for minimizing the effects of noise is called digital signal processing. Engineers are more concerned about internal noise at high frequencies than at low frequencies, because the less external noise there is, the more significant the internal noise becomes.The various types of noise are given as follows :

2.1.1 External Noise

The various forms of noise created outside the receiver come under the heading of external noise and include atmospheric and extraterrestrial noise and industrial noise are explained in details below:

2.1.1.1 Atmospheric Noise

Atmospheric noise is radio noise caused by natural atmospheric processes, primarily lightning discharges in thunderstorms. Atmospheric noise is mainly caused by cloud-to-ground flashes as the current is much stronger than for cloud-to-cloud flashes. On a worldwide scale, eight million lightning flashes occur daily. This is about 100 lightning flashes per second. The sum of all these lightning flashes results in atmospheric noise. It can be observed with a radio receiver in the form of a combination of white noise (coming from distant thunderstorms) and impulse noise (coming from a near thunderstorm). The power-sum varies with seasons and nearness of thunderstorm centers. Although lightning has a broad-spectrum emission, its noise power increases with decreasing frequency. Therefore, at very low frequency and low frequency, atmospheric noise often dominates, while at high Frequency, man-made noise dominates in rural areas. From 1960s to 1980s, a worldwide effort was made to measure the atmospheric noise and variations.

2.1.1.2 Extraterrestrial Noise

It is safe to say that there are almost as many types of space noise as there are sources. For convenience, a division into two subgroups will suffice solar noise. The sun radiates so many things our way that we should not be too surprised to find that noise is noticeable among them, again there are two types. Under normal "quiet" conditions, there is a constant noise radiation from the sun, simply because it is a large body at a very high temperature (over 6OOO°C on the surface). It therefore radiates over a very broad frequency spectrum, which includes the frequencies we use for communications. However, the sun is a constantly changing star, which undergoes cycles of peak activity from which electrical disturbances erupt, such as corona flares and sunspots. Even though the additional noise produced comes from a limited portion of the sun's surface, it may still be orders of magnitude greater than that received during periods of quiet sun.

2.1.1.3 Industrial Noise

Industrial noise is usually considered mainly from the point of view of environmental health and safety, rather than nuisance, as sustained exposure can cause permanent hearing damage. Traditionally, workplace noise has been a hazard linked to heavy industries such as ship-building and associated only with noise induced hearing loss (NIHL). Modern thinking in occupational safety and health identifies noise as hazardous to worker safety and health in many places of employment and by a variety of means. Noise can not only cause hearing impairment (at long-term exposures of over 85 decibels (dB), known as an exposure action value), but it also acts as a causal factor for stress and raises systolic blood pressure. Additionally, it can be a causal factor in work accidents, both by masking hazards and warning signals, and by impeding concentration. Noise also acts synergistically with other hazards to increase the risk of harm to workers. In particular, noise and dangerous substances (e.g. some solvents) that have some tendencies towards ototoxicity may give rise to rapid ear damage. A-weighted measurements are commonly used to determine noise levels that can cause harm to the human ear, and special exposure meters are available that integrate noise over a period of time to give an equivalent value (equivalent sound pressure level), defined by standards. Between the frequencies of 1 to 600 MHz (in urban, suburban and other industrial areas) the intensity of noise made by humans easily outstrips that created by any other source,

internal or external to the receiver. The nature of industrial noise is so variable that it is difficult to analyze it on any basis other than the statistical. It does, however, obey the general principle that received noise increases as the receiver bandwidth is increased.

2.1.2 Internal Noise

Internal noise represents all the different types that arise inside of the communication system components. It includes thermal noise, shot noise, and flicker noise. Although the components of both the transmitter and receiver are included in the definition, the region of primary concern is from the receiving antenna through the first several stages of the receiver. It is in this region of small signal amplitudes that internal noise is most troublesome.

2.1.2.1 Thermal Noise

Johnson–Nyquist noise (thermal noise, Johnson noise, or Nyquist noise) is the electronic noise generated by the thermal agitation of the charge carriers (usually the electrons) inside an electrical conductor at equilibrium, which happens regardless of any applied voltage. Thermal noise is approximately white, meaning that the power spectral density is nearly equal throughout the frequency spectrum. Additionally, the amplitude of the signal has very nearly Gaussian Probability Density Function. This type of noise was first measured by John B. Johnson at Bell labs in 1928.[1]

Noise Voltage and Power

Thermal noise is distinct from shot noise, which consists of additional current fluctuations that occur when a voltage is applied and a macroscopic current starts to flow. For the general case, the above definition applies to charge carriers in any type of conducting medium (e.g. ions in an electrolyte), not just resistors. It can be modelled by a voltage source representing the noise of the non-ideal resistor in series with an ideal noise free resistor. The power spectral density, or voltage variance (mean square) per hertz of bandwidth, is given by

$$\bar{v_n^2} = 4k_B TR \qquad (2.1)$$

where k_B is Boltzmann's constant in joules per kelvin, T is the resistor's absolute temperature in kelvins, and R is the resistor value in ohms (Ω). A resistor in a short circuit dissipates a noise power given by:

$$P = \bar{v_n^2}/R = 4k_B\, T\Delta f \qquad (2.2)$$

The noise generated at the resistor can transfer to the remaining circuit; the maximum noise power transfer happens with impedance matching when the Thevenin equivalent resistance of the remaining circuit is equal to the noise generating resistance. In this case each one of the two participating resistors dissipates noise in both itself and in the other resistor. Since only half of the source voltage drops across any one of these resistors, the resulting noise power is given by

$$P = k_B T \Delta f \tag{2.3}$$

Where, P is the thermal noise power. It is often measured in decibels relative to 1 milliwatt (dBm), assuming a 50 ohm load resistance. With these conventions, thermal noise for a resistor at room temperature can be estimated as:

$$P_{\mathrm{dBm}} = -174 + 10\ \log_{10}(\Delta f_{\mathrm{Hz}}) \tag{2.4}$$

where P is measured in dBm. The actual amount of thermal noise received by a radio receiver having a 50 Ω input impedance, connected to an antenna with a 50 Ω radiation resistance would be scaled by the noise figure (NF), shown as follows:

$$P_{\mathrm{receivernoise}}(dB) = P_{\mathrm{resistornoise}}(dB) + 10\ \log_{10}(10^{NF/10} - 1) \tag{2.5}$$

Here $P_{\mathrm{receivernoise}}$ is the noise generated in the receiver itself. $P_{\mathrm{resistornoise}}$ is the value from the table above for thermal noise from a resistor. NF is in dB. Ten raised to the power of NF/10 is called the noise factor and is simply the linear version of noise figure. We subtract one from the noise factor here because a noise factor of 1 is a perfect receiver (which contributes no noise to the signal). Noise factor is defined this way because in the laboratory, it is measured by connecting a resistor to the receiver's input and comparing the output noise to what one would expect if the noise of the resistor were simply amplified by the gain of the receiver. A ratio of 1 between the actual output noise in the numerator and the gain times the resistor thermal noise in the denominator means that the receiver added no noise of its own to the input.

The radiation resistance of the antenna does not convert power to heat, and so is not a source of thermal noise. It dissipates power as electromagnetic radiation. Thus if the radiation incident on the antenna is blackbody radiation at the same temperature as the antenna then it will contribute noise in the same manner based on the radiation resistance, according to the zeroth law of thermodynamics as applied between the surrounding

matter, the antenna, and the electromagnetic radiation. In an underground cave or mine, the antenna will be surrounded by earth which is opaque to the radiation, and thus, the radiation resistance will contribute to the thermal noise. Likewise, the load impedance of the input of the receiver does not contribute directly to received noise. Therefore, it is indeed possible, and even common, for a receiver to have a noise factor of less than 2X (or equivalently, a noise figure of less than 3 dB). For example, a 6 MHz wide channel such as a television channel received signal would compete with the tiny amount of power generated by room temperature in the input stages of the receiver, which, for a TV receiver with a noise figure of 3 dB would be −106 dBm, or one fortieth of a picowatt. For a TV with a noise figure of 1 dB, the noise power would be −112 dBm. The actual source of this noise is a combination of thermal noise in physical resistances of wires and semiconductors, thermal noise in other lossy devices such as transformers, as well as shot noise.The 6 MHz bandwidth could be the 6 MHz between 54 and 60 MHz (corresponding to TV channel 2) or the 6 MHz between 470 MHz and 476 MHz (corresponding to TV channel UHF 14) or any other 6 MHz in the spectrum for that matter. The bandwidth of any channel should never be confused with the transmitting frequency of a channel. For example, a channel transmit frequency may be as high as 2450 MHz for a WIFI signal, but the actual width of the channel may be only 20 MHz, and that 20 MHz would be the correct value to use in computing the Johnson–Nyquist noise. It is quite possible to detect a signal whose amplitude is less than the noise contained within its bandwidth. The Global Positioning System (GPS) and Glonass system both have signal amplitudes that are less than the received noise in a typical receiver at ground level. In the case of GPS, the received signal has a power of −133 dBm. The newer batch of satellites have a more powerful transmitter. To achieve this feat, GPS uses spread spectrum techniques, while some other communication systems use error control coding.

Noise Current

The noise source can also be modeled by a current source in parallel with the resistor by taking the Norton equivalent that corresponds simply to divide by R.

This gives the root mean square value of the current source as:

$$i_n = \sqrt{\frac{4k_B T \Delta f}{R}} \tag{2.6}$$

Thermal noise is intrinsic to all resistors and is not a sign of poor design or manufacture, although resistors may also have excess noise.

2.1.2.2 Shot Noise

Shot noise is a type of electronic noise that occurs when the finite number of particles that carry energy, such as electrons in an electronic circuit or photons in an optical device, is small enough to give rise to detectable statistical fluctuations in a measurement. It is important in electronics, telecommunications, and fundamental physics. It also refers to an analogous noise in particle simulations, where due to the small number of particles, the simulation exhibits detectable statistical fluctuations not observed in the real-world system. The magnitude of this noise increases with the average magnitude of the current or intensity of the light. However, since the magnitude of the average signal increases more rapidly than that of the shot noise (its relative strength decreases with increasing signal), shot noise is often only a problem with small currents or light intensities. The intensity of a source will yield the average number of photons collected, but knowing the average number of photons which will be collected will not give the actual number collected. The actual number collected will be more than, equal to, or less than the average, and their distribution about that average will be a Poisson distribution. Since the Poisson distribution approaches a normal distribution for large numbers, the photon noise in a signal will approach a normal distribution for large numbers of photons collected. The standard deviation of the photon noise is equal to the square root of the average number of photons. The signal-to-noise ratio is then given by:

$$\mathrm{SNR} = \frac{N}{\sqrt{N}} = \sqrt{N} \tag{2.7}$$

where N is the average number of photons collected. When N is very large, the signal-to-noise ratio is very large as well. It can be seen that photon noise becomes more important when the number of photons collected is small. Shot noise exists because phenomena such as light and electrical current consist of the movement of discrete, quantized 'packets'. Imagine light coming out of a laser pointer and hitting a wall. That light comes in small packets, or photons. When the spot is bright enough to see, there are many

billions of light photons that hit the wall per second. Now, imagine turning down the laser brightness until the laser is almost off. Then, only a few photons hit the wall every second. But the fundamental physical processes that govern light emission say that these photons are emitted from the laser at random times. So if the average number of photons that hit the wall each second is 5, some seconds when you measure you will find 2 photons, and some 10 photons. These fluctuations are shot noise. Low noise active electronic devices are designed such that shot noise is suppressed by the electrostatic repulsion of the charge carriers. Space charge limiting is not possible in photon devices.

2.1.3 White Noise

White noise is a random signal (or process) with a flat power spectral density. In other words, the signal contains equal power within a fixed bandwidth at any center frequency. White noise draws its name from white light in which the power spectral density of the light is distributed over the visible band in such a way that the eye's three color receptors (cones) are approximately equally stimulated. An infinite-bandwidth white noise signal is purely a theoretical construction. By having power at all frequencies, the total power of such a signal is infinite and therefore impossible to generate. In practice, however, a signal can be "white" with a flat spectrum over a defined frequency band.

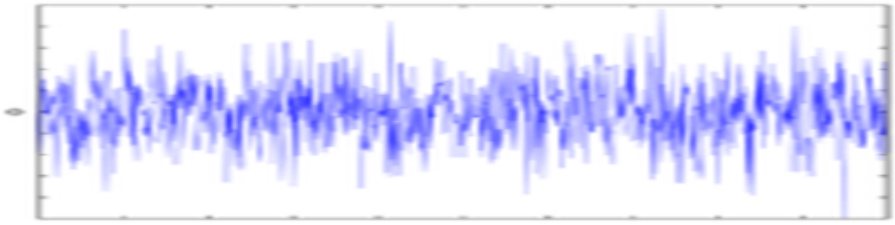

Figure 2.1 Realization of a Gaussian White Noise Process

The image to the right displays a finite length, discrete time realization of a white noise process generated from a computer. Being uncorrelated in time does not restrict the values a signal can take. Any distribution of values is possible (although it must have zero DC component). Even a binary signal which can only take on the values 1 or -1 will be white if the sequence is statistically uncorrelated. Noise having a continuous distribution, such as a normal distribution, can of course be white. It is often incorrectly assumed that Gaussian noise, i.e., noise with a Gaussian amplitude is necessarily white noise, yet neither property implies the other. Gaussian white noise is a good approximation of many real-world situations and generates mathematically tractable

models. Gaussian white noise as the useful statistical property that its values are independent. White noise is the generalized mean-square derivative of the Wiener process or Brownian motion.

Mathematical Definition

A random vector **w** is a white random vector if and only if its mean vector and autocorrelation matrix are the following:

$$\mu_w = \mathbb{E}\{\mathbf{w}\} = 0 \tag{2.8}$$

$$R_{ww} = \mathbb{E}\{\mathbf{w}\mathbf{w}^T\} = \sigma^2\mathbf{I}. \tag{2.9}$$

That is, it is a zero mean random vector, and its autocorrelation matrix is a multiple of the identity matrix. When the autocorrelation matrix is a multiple of the identity, we say that it has spherical correlation.

White Random Process

A continuous time random process $w(t)$ where is a white noise process if and only if its mean function and autocorrelation function satisfy the following:

$$\mu_w(t) = \mathbb{E}\{w(t)\} = 0 \tag{2.10}$$

$$R_{ww}(t_1, t_2) = \mathbb{E}\{w(t_1)w(t_2)\} = (N_0/2)\delta(t_1 - t_2) \tag{2.11}$$

i.e. it is a zero mean process for all time and has infinite power at zero time shift since its autocorrelation function is the Dirac delta function. The above autocorrelation function implies the following power spectral density given by:

$$S_{xx}(\omega) = N_0/2, \tag{2.12}$$

Since the Fourier transform of the delta function and likewise is equal to 1. Since this power spectral density is the same at all frequencies, call it white as an analogy to the frequency spectrum of white light. A generalization to random elements on infinitely dimensional spaces, such as random fields, is the white noise measure. Two theoretical applications using a white random vector are the simulation and whitening of another arbitrary random vector. To simulate an arbitrary random vector, transform a white random vector with a carefully chosen matrix. Choose the transformation matrix so that the mean and covariance matrix of the transformed white random vector matches the mean and covariance matrix of the arbitrary random vector that are simulating. To whiten an arbitrary random vector, transform it by a different carefully chosen matrix so that the output random vector is a white random vector. These two ideas are crucial in

applications such as channel estimation and channel equalization in communications and audio. These concepts are also used in data compression. Suppose that a random vector X has covariance matrix K_{xx}. Since this matrix is Hermitian symmetric and positive semidefinite, by the spectral theorem from linear algebra, diagonalize or factor the matrix in the following way.

$$K_{xx} = E\Lambda E^T \tag{2.13}$$

where E is the orthogonal matrix of eigenvectors and Λ is the diagonal matrix of eigenvalues. This can simulate the 1st and 2nd moment properties of this random vector with mean μ and covariance matrix K_{xx} via the following transformation of a white vector **w**:

$$\mathbf{x} = H\,\mathbf{w} + \mu \tag{2.14}$$

where

$$H = E\Lambda^{1/2} \tag{2.15}$$

Thus, the output of this transformation has expectation

$$\mathbb{E}\{\mathbf{x}\} = H\,\mathbb{E}\{\mathbf{w}\} + \mu = \mu \tag{2.16}$$

The method for whitening a vector **x** with mean μ and covariance matrix K_{xx} is to perform the following calculation:

$$\mathbf{w} = \Lambda^{-1/2} E^T (\mathbf{x} - \mu) \tag{2.17}$$

Thus, the output of this transformation has expectation

$$\mathbb{E}\{\mathbf{w}\} = \Lambda^{-1/2} E^T (\mathbb{E}\{\mathbf{x}\} - \mu) = \Lambda^{-1/2} E^T (\mu - \mu) = 0 \tag{2.18}$$

and covariance matrix

$$\mathbb{E}\{\mathbf{w}\mathbf{w}^T\} = \mathbb{E}\{\Lambda^{-1/2} E^T (\mathbf{x} - \mu)(\mathbf{x} - \mu)^T E\,\Lambda^{-1/2}\} \tag{2.19}$$

By diagonalizing K_{xx}, we get the following:

$$\Lambda^{-1/2} E^T E\Lambda E^T E\,\Lambda^{-1/2} = \Lambda^{-1/2}\,\Lambda\,\Lambda^{-1/2} = I \tag{2.20}$$

Thus, with the above transformation, this will whiten the random vector to have zero mean and the identity covariance matrix.

The same two concepts of simulating and whitening cannot be extended to the case of continuous time random signals or processes. For simulating, create a filter into which feed a white noise signal. Choose the filter so that the output signal simulates the 1st and

2nd moments of any arbitrary random process. For whitening, feed any arbitrary random signal into a specially chosen filter so that the output of the filter is a white noise signal.

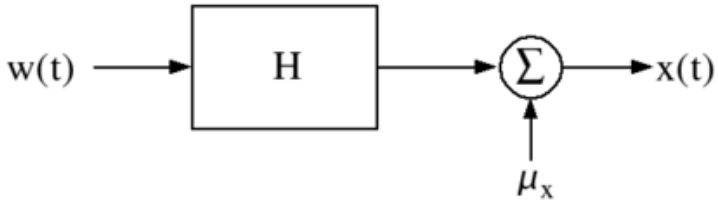

Figure 2.2 White Noise fed into a Linear Time-Invariant Filter

From above figure simulate any wide-sense stationary, continuous-time random process with constant mean and covariance function as follows:

$$K_x(\tau) = \mathbb{E}\left\{(x(t_1) - \mu)(x(t_2) - \mu)^*\right\} \text{ where } \tau = t_1 - t_2 \quad (2.21)$$

and power spectral density is given by:

$$S_x(\omega) = \int_{-\infty}^{\infty} K_x(\tau)\, e^{-j\omega\tau}\, d\tau \quad (2.22)$$

Because $K_x(\tau)$ is Hermitian symmetric and positive semi-definite, it follows that $S_x(\omega)$ is real and can be factored as

$$S_x(\omega) = |H(\omega)|^2 = H(\omega)\, H^*(\omega) \quad (2.23)$$

if and only if $S_x(\omega)$ satisfies the Paley-Wiener criterion.

$$\int_{-\infty}^{\infty} \frac{\log(S_x(\omega))}{1+\omega^2}\, d\omega < \infty \quad (2.24)$$

If $S_x(\omega)$ is a rational function, factor it into pole-zero form as

$$S_x(\omega) = \frac{\Pi_{k=1}^{N}(c_k - j\omega)(c_k^* + j\omega)}{\Pi_{k=1}^{D}(d_k - j\omega)(d_k^* + j\omega)} \quad (2.25)$$

Choosing a minimum phase $H(\omega)$ so that its poles and zeros lie inside the left half s-plane, then simulate x(t) with $H(\omega)$ as the transfer function of the filter. Simulate x(t) by constructing the following linear, time-invariant filter:

$$\hat{x}(t) = \mathcal{F}^{-1}\left\{H(\omega)\right\} * w(t) + \mu \quad (2.26)$$

where $w(t)$ is a continuous-time, white-noise signal with the following 1st and 2nd moment properties:

$$\mathbb{E}\{w(t)\} = 0 \quad (2.27)$$

$$\mathbb{E}\{w(t_1)w^*(t_2)\} = K_w(t_1, t_2) = \delta(t_1 - t_2) \quad (2.28)$$

Thus, the resultant signal has the same 2nd moment properties as the desired signal x(t). For whitening a continuous-time random signal using frequency domain techniques, factor the power spectral density $S_x(\omega)$ as described. Choosing the minimum phase $H(\omega)$ so that its poles and zeros lie inside the left half s-plane, then whiten x(t) with the following inverse filter:

$$H_{inv}(\omega) = \frac{1}{H(\omega)}. \tag{2.29}$$

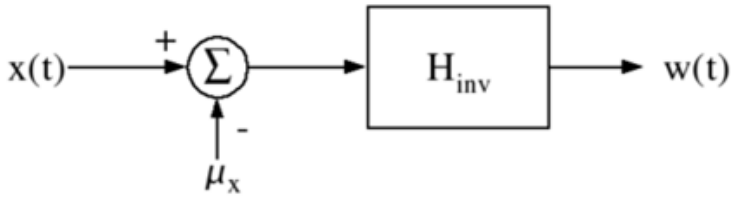

Figure 2.3 Arbitrary Random Process x(t) fed into Time-Invariant filter

Choose the minimum phase filter so that the resulting inverse filter is stable. The final form of the whitening procedure is as follows:

$$w(t) = \mathcal{F}^{-1}\{H_{inv}(\omega)\} * (x(t) - \mu) \tag{2.30}$$

so that w(t) is a white noise random process with zero mean and constant, unit power spectral density

$$S_w(\omega) = \mathcal{F}\{\mathbb{E}\{w(t_1)w(t_2)\}\} = H_{inv}(\omega)S_x(\omega)H^*_{inv}(\omega) = \frac{S_x(\omega)}{S_x(\omega)} = 1. \tag{2.31}$$

2.1.4 Miscellaneous Noise

2.1.4.1 Comic Noise

Cosmic noise and galactic radio noise is random noise that originates outside the Earth's atmosphere. It can be detected and heard on radio receivers. Cosmic noise characteristics are similar to those of thermal noise. Cosmic noise is experienced at frequencies above about 15 MHz when highly directional antennas are pointed toward the sun or to certain other regions of the sky such as the center of the Milky Way Galaxy. Celestial objects like Quasars, super dense objects that lie far from earth, emit electromagnetic waves in its full spectrum including radio waves. Cosmic Microwave Background Radiation (CMBR) from outer space, CMBR, thought to be a relic of the big bang, pervades the space almost homogeneously from all around. In this particular instance, electromagnetic spectrum is wide, though the peak is in the microwave range.

2.1.4.2 Gaussian Noise

Gaussian noise is statistical noise that has a probability density function of the normal distribution (also known as Gaussian distribution). In other words, the values that the noise can take on are Gaussian-distributed. It is most commonly used as additive white noise to yield additive white Gaussian noise (AWGN). Gaussian noise is properly defined as the noise with a Gaussian amplitude distribution. This says nothing of the correlation of the noise in time or of the spectral density of the noise. Labeling Gaussian noise as 'white' describes the correlation of the noise.

2.1.4.3 Grey Noise

Grey noise is random noise subjected to a psychoacoustic equal loudness curve (such as an inverted A-weighting curve) over a given range of frequencies, giving the listener the perception that it is equally loud at all frequencies. This is in contrast to white noise, noise which is in fact equally loud at all frequencies but not perceived as such due to psychoacoustics.

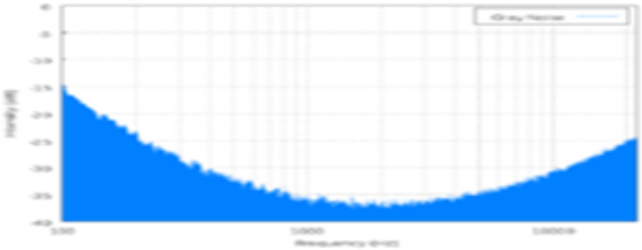

Figure 2.4 Grey Noise Spectrum

2.1.4.4 Burst Noise

Burst noise is a type of electronic noise that occurs in semiconductors. It is also called popcorn noise, impulse noise, bi-stable noise, or random telegraph signal (RTS) noise. It consists of sudden step-like transitions between two or more discrete voltage or current levels, as high as several hundred micro volts, at random and unpredictable times. Each shift in offset voltage or current often lasts from several milliseconds to seconds, and sounds like popcorn popping if hooked up to an audio speaker [20].

Popcorn noise was first observed in early point contact diodes, then re-discovered during the commercialization of one of the first semiconductor op-amps. No single source of popcorn noise is theorized to explain all occurrences, however the most commonly invoked cause is the random trapping and release of charge

carriers at thin film interfaces or at defect sites in bulk semiconductor crystal. In cases where these charges have a significant impact on transistor performance (such as under an MOS gate or in a bipolar base region), the output signal can be substantial. These defects can be caused by manufacturing processes, such as heavy ion implantation, or by unintentional side-effects such as surface contamination. Individual op-amps can be screened for popcorn noise with peak detector circuits, to minimize the amount of noise in a specific application [20].

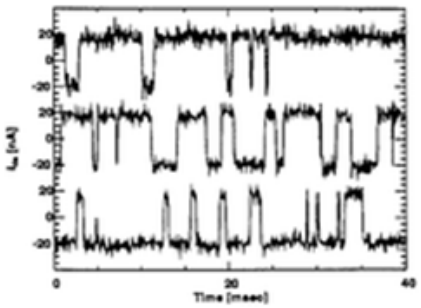

Figure 2.5 Graph of Burst Noise

2.1.4.5 Flicker Noise

Flicker noise is a type of electronic noise with a $1/f$, or pink spectrum. It is therefore often referred to as $1/f$ noise or pink noise, though these terms have wider definitions. It occurs in almost all electronic devices, and results from a variety of effects, such as impurities in a conductive channel, generation and recombination noise in a transistor due to base current, and so on. It is always related to a direct current. In electronic devices, it is a low-frequency phenomenon, as the higher frequencies are overshadowed by white noise from other sources. In oscillators, however, the low-frequency noise is mixed up to frequencies close to the carrier which results in oscillator phase noise. Flicker noise is often characterized by the corner frequency f_c between the regions dominated by each type. MOSFETs have a higher f_c between than JFETs or bipolar transistors which is usually below 2 kHz for the latter. The flicker noise voltage power in MOSFET can be expressed by K/(Cos•WLf), where K is the process-dependent constant, W and L are channel width and length respectively[15]. Flicker noise is found in carbon composition resistors, where it is referred to as excess noise, since it increases the overall noise level above the thermal noise level, which is present in all resistors. In contrast, wire-wound resistors have the least amount of flicker noise. Since flicker noise is related to the level

of DC, if the current is kept low, thermal noise will be the predominant effect in the resistor, and the type of resistor used will not affect noise levels.

2.2 MEASUREMENT

For measurements the interest is in the "drift" of a variable with respect to a measurement at a previous time. This is calculated by applying the signal time differencing:

$$1 - e^{-Td \cdot s} \tag{2.32}$$

After some manipulation, the variance of the voltage difference is:

$$2a \int_0^{f_h} \frac{1 - \cos(Td \cdot 2\pi f)}{f} df = -2a \cdot Cin(Td \cdot 2\pi f_h). \tag{2.33}$$

[21] where f_h is a brick-wall filter limiting the upper bandwidth during measurement.

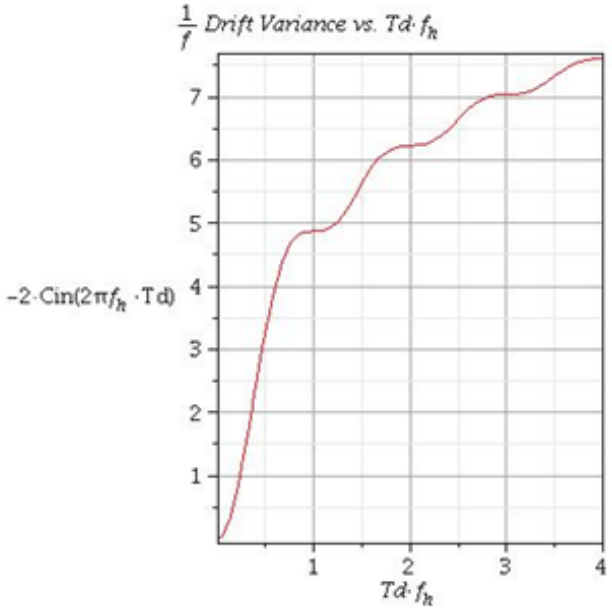

Figure 2.6 Real Measurements Involve More Complicated Calculations

2.3 QUANTIZATION ERROR (NOISE)

The difference between the actual analog value and quantized digital value due is called quantization error. This error is due either to rounding or truncation.

2.3.1 Quantization Noise Model of Quantization Error

The difference between the blue and red signals in the below figure is the quantization error, which is "added" to the quantized signal and is the source of noise.

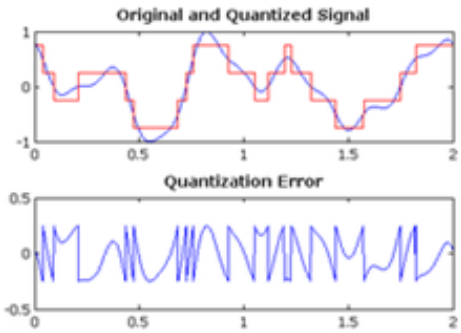

Figure 2.7 Quantization Noise Model of Quantization Error

Quantization noise is a model of quantization error introduced by quantization in the analog-to-digital conversion (ADC) in telecommunication systems and signal processing. It is a rounding error between the analog input voltage to the ADC and the output digitized value. The noise is non-linear and signal-dependent. It can be modelled in several different ways. In an ideal analog-to-digital converter, where the quantization error is uniformly distributed between −1/2 LSB and +1/2 LSB, and the signal has a uniform distribution covering at all quantization levels, the signal-to-noise ratio (SNR) can be calculated from

$$\mathrm{SNR}_{\mathrm{ADC}} = 20\log_{10}(2^Q) \approx 6.0206 \cdot Q \text{ dB} \qquad (2.34)$$

The most common test signals that fulfil this are full amplitude triangle waves and sawtooth waves. In this case a 16-bit ADC has a maximum signal-to-noise ratio of 6.0206 × 16 = 96.33 dB. When the input signal is a full-amplitude sine wave the distribution of the signal is no longer uniform, and the corresponding equation is instead given by:

$$\mathrm{SNR}_{\mathrm{ADC}} \approx 1.761 + 6.0206 \cdot Q \text{ dB} \qquad (2.35)$$

Here, the quantization noise is once again assumed to be uniformly distributed. When the input signal has a high amplitude and a wide frequency spectrum this is the case.[21] In this case a 16-bit ADC has a maximum signal-to-noise ratio of 98.09 dB. The 1.761 difference in signal-to-noise only occurs due to the signal being a full-scale sine wave instead of a triangle/saw tooth. For complex signals in high-resolution ADCs this is an accurate model. For low-resolution ADCs, low-level signals in high-resolution ADCs, and for simple waveforms the quantization noise is not uniformly distributed, making this model inaccurate. In these cases, the quantization noise distribution is strongly affected by the exact amplitude of the signal. The calculations above, however, assume a

completely filled input channel. If this is not the case - if the input signal is small - the relative quantization distortion can be very large.

Quantization Error = Quantized Output - Analog Input

2.4 ADDITIVE WHITE GAUSSIAN NOISE

In communications, the additive white Gaussian noise (AWGN) channel model is one in which the only impairment is a linear addition of wideband or white noise with a constant spectral density (expressed as watts per hertz of bandwidth) and a Gaussian distribution of amplitude. The model does not account for the phenomena of fading, frequency selectivity, interference, nonlinearity or dispersion. However, it produces simple and tractable mathematical models which are useful for gaining insight into the underlying behavior of a system before these other phenomena are considered.

Wideband Gaussian noise comes from many natural sources, such as the thermal vibrations of atoms in antennas shot noise, black body radiation from the earth and other warm objects, and from celestial sources such as the Sun. The AWGN channel is a good model for many satellite and deep space communication links. It is not a good model for most terrestrial links because of multipath, terrain blocking, interference, etc. However, for terrestrial path modeling, AWGN is commonly used to simulate background noise of the channel under study, in addition to multipath, terrain blocking, interference, ground clutter and self interference that modern radio systems encounter in terrestrial operation.

2.5 NOISE CALCULATIONS

2.5.1 Signal-To-Noise-Ratio

Signal-to-noise ratio (often abbreviated SNR or S/N) is an electrical engineering measurement, also used in other fields (such as scientific measurement or biological cell signaling), defined as the ratio of a signal power to the noise power corrupting the signal. A ratio higher than 1:1 indicates more signal than noise. In less technical terms, signal-to-noise ratio compares the level of a desired signal (such as music) to the level of background noise. The higher the ratio, the less obtrusive the background noise is.

2.5.2 Technical Sense

In engineering, signal-to-noise ratio is a term for the power ratio between a signal (meaningful information) and the background noise:

$$\mathrm{SNR} = \frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}}, \tag{2.36}$$

where P is average power. Both signal and noise power must be measured at the same or equivalent points in a system, and within the same system bandwidth. If the signal and the noise are measured across the same impedance, then the SNR can be obtained by calculating the square of the amplitude ratio.

$$\mathrm{SNR} = \frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}} = \left(\frac{A_{\mathrm{signal}}}{A_{\mathrm{noise}}}\right)^2, \tag{2.37}$$

where A is root mean square (RMS) amplitude (for example, typically, RMS voltage). Because many signals have a very wide dynamic range, SNRs are usually expressed in terms of the logarithmic decibel scale. In decibels, the SNR is, by definition, 10 times the logarithm of the power ratio:

$$\mathrm{SNR(dB)} = 10\log_{10}\left(\frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}}\right) = 20\log_{10}\left(\frac{A_{\mathrm{signal}}}{A_{\mathrm{noise}}}\right) = P_{\mathrm{signal,dB}} - P_{\mathrm{noise,dB}}. \tag{2.38}$$

A common alternative definition of SNR is the ratio of mean to standard deviation of a signal or measurement:[9]

$$\mathrm{SNR} = \mu / \sigma \tag{2.39}$$

where is the signal, or the mean or expected value of the signal, or some measure of signal strength, and σ is the standard deviation of the noise, or an estimate thereof. The exact methods may vary between fields. For example, if the signal data are known to be constant, then σ can be calculated using the standard deviation of the signal. If the signal data are not constant, then σ can be calculated from data where the signal is zero or relatively constant.

2.5.3 Electrical SNR and Acoustics

Often the signals being compared are electromagnetic in nature, though it is also possible to apply the term to sound stimuli. Due to the definition of decibel, the SNR gives the same result independent of the type of signal which is evaluated (such as power, current, or voltage). Signal-to-noise ratio is closely related to the concept of dynamic range, where dynamic range measures the ratio between noise and the greatest un-distorted signal on a channel. SNR measures the ratio between noise and an arbitrary signal on the channel, not necessarily the most powerful signal possible. Because of this, measuring signal-to-noise ratios requires the selection of a representative or reference signal. In

audio engineering, this reference signal is usually a sine wave, sounding a tone, at a recognized and standardized nominal level or alignment level, such as 1 kHz at +4 dBu (1.228 VRMS). SNR is usually taken to indicate an average signal-to-noise ratio, as it is possible that (near) instantaneous signal-to-noise ratios will be considerably different. The concept can be understood as normalizing the noise level to 1 (0 dB) and measuring how far the signal 'stands out'. In general, higher signal to noise is better; the signal is 'cleaner'.

In image processing, the SNR of an image is usually defined as the ratio of the mean pixel value to the standard deviation of the pixel values, that is, SNR = μ / σ (the inverse of the coefficient of variation). Related measures are the "contrast ratio" and the "contrast-to-noise ratio". The connection between optical power and voltage in an imaging system is linear. This usually means that the SNR of the electrical signal is calculated by the 10 log rule. With an interferometric system, however, where interest lies in the signal from one arm only, the field of the electromagnetic wave is proportional to the voltage (assuming that the intensity in the second, the reference arm is constant). Therefore the optical power of the measurement arm is directly proportional to the electrical power and electrical signals from optical inteferrometry are following the 20 log rule. The Rose criterion (named after Albert Rose) states that an SNR of at least 5 is needed to be able to distinguish image features at 100% certainty. An SNR less than 5 means less than 100% certainty in identifying image details.[19]

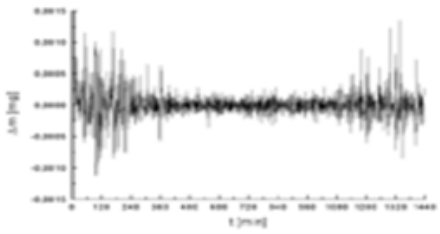

Figure 2.8 Recording of the Noise of a Thermogravimetric

In above figure 2.8, for recording of the noise of a thermogravimetric analysis device that is poorly isolated from a mechanical point of view; the middle of the curve shows a lower noise, due to a lesser surrounding human activity at night. Any measurement device is

disturbed by parasitic phenomena. This includes the electronic noise as described above, but also any external event that affects the measured phenomenon — wind, vibrations, gravitational attraction of the moon, variations of temperature, variations of humidity, etc. depending on what is measured and of the sensitivity of the device. It is often possible to reduce the noise by controlling the environment. Otherwise, when the characteristics of the noise are known and are different from the signal, it is possible to filter it or to process the signal. When the noise is a random perturbation and the signal is a constant value, it is possible to enhance the SNR by increasing the measurement time. Similarly, non-constant signals may be enhanced by calculating the average of several repeated measurements.

2.6 DIGITAL SIGNALS

When using digital storage the number of bits of each value determines the maximum signal-to-noise ratio. In this case, the noise is the error signal caused by the quantization of the signal, taking place in the analog-to-digital conversion. The noise level is non-linear and signal-dependent; different calculations exist for different signal models. The noise is modeled as an analog error signal being summed with the signal before quantization ("additive noise").The modulation error ratio (MER) is a measure of the SNR in a digitally modulated signal. Like SNR, MER can be expressed in dB.

2.7 FIXED POINT

For n-bit integers with equal distance between quantization levels (uniform quantization) the dynamic range (DR) is also determined. Assuming a uniform distribution of input signal values, the quantization noise is a uniformly-distributed random signal with a peak-to-peak amplitude of one quantization level, making the amplitude ratio 2n/1. The formula is then:

$$\mathrm{DR(dB)} = \mathrm{SNR(dB)} = 20\log_{10}(2^n) \approx 6.02 \cdot n \qquad (2.40)$$

This relationship is the origin of statements like "16-bit audio has a dynamic range of 96 dB". Each extra quantization bit increases the dynamic range by roughly 6 dB Assuming a full-scale sine wave signal (that is, the quantizer is designed such that it has the same minimum and maximum values as the input signal), the quantization noise approximates a sawtooth wave with peak-to-peak amplitude of one quantization level[25] and uniform distribution. In this case, the SNR is approximately:

$$\mathrm{SNR(dB)} \approx 20\log_{10}(2^n\sqrt{3/2}) \approx 6.02 \cdot n + 1.761 \qquad (2.41)$$

2.8 FLOATING POINT

Floating-point numbers provide a way to trade off signal-to-noise ratio for an increase in dynamic range. For n bit floating-point numbers, with n-m bits in the mantissa and m bits in the exponent:

$$\mathrm{DR(dB)} = 6.02 \cdot 2^m \qquad (2.42)$$

$$\mathrm{SNR(dB)} = 6.02 \cdot (n-m) \qquad (2.43)$$

The dynamic range is much larger than fixed-point, but at a cost of a worse signal-to-noise ratio. This makes floating-point preferable in situations where the dynamic range is large or unpredictable. Fixed-point's simpler implementations can be used with no signal quality disadvantage in systems where dynamic range is less than 6.02m. The very large dynamic range of floating-point can be a disadvantage, since it requires more forethought in designing algorithms.[25]

2.9 NOISE FIGURE

Noise figure (NF) is a measure of degradation of the signal to noise ratio (SNR), caused by components in the RF signal chain. The noise figure is the ratio of the output noise power of a device to the portion thereof attributable to thermal noise in the input termination at standard noise temperature (usually 290 K). The noise figure is thus the ratio of actual output noise to that which would remain if the device itself did not introduce noise. It is a number by which the performance of a radio receiver can be specified. In heterodyne systems, output noise power includes spurious contributions from image-frequency transformation, but the portion attributable to thermal noise in the input termination at standard noise temperature includes only that which appears in the output via the principal frequency transformation of the system and excludes that which appears via the image frequency transformation. Essentially, the noise figure is the difference in decibels (dB) between the noise output of the actual receiver to the noise output of an “ideal” receiver with the same overall gain and bandwidth when the receivers are connected to sources at the standard noise temperature T_0 (usually 290 K). The noise power from a simple load is equal to kTB, where k is Boltzmann's constant, T is the absolute temperature of the load and B is the measurement bandwidth. This makes the noise figure a useful figure of merit for terrestrial systems where the antenna effective

temperature is usually near the standard 290 K. In this case, one receiver with a noise figure say 2 dB better than another, will have an output signal to noise ratio that is about 2 dB better than the other. However, in the case of satellite communications systems, where the antenna is pointed out into cold space, the antenna effective temperature is often colder than 290 K. In these cases a 2 dB improvement in receiver noise figure will result in more than a 2 dB improvement in the output signal to noise ratio. For this reason, the related figure of effective noise temperature is therefore often used instead of the noise figure for characterizing satellite-communication receivers and LNA.

2.9.1 Definition

The noise figure of a system can be defined as

$$NF = \frac{\mathrm{SNR_{in}}}{\mathrm{SNR_{out}}} \tag{2.44}$$

where $\mathrm{SNR_{in}}$ and $\mathrm{SNR_{out}}$ are the input and output signal-to-noise ratios, respectively. Alternatively, noise figure may be defined in terms of dB units

$$NF_{dB} = 10 \log \left(\frac{\mathrm{SNR_{in}}}{\mathrm{SNR_{out}}} \right) = \mathrm{SNR_{in,dB}} - \mathrm{SNR_{out,dB}} \tag{2.45}$$

where $\mathrm{SNR_{in,dB}}$ and $\mathrm{SNR_{out,dB}}$ are in dB. The previous formula is only valid when the input termination is at standard noise temperature T_0.These definitions are equivalent, differing only in the use of dB units. The first definition is sometimes referred to as noise factor to distinguish it from the dB form. The noise factor of a device is related to its noise temperature via

$$F = 1 + \frac{T_\mathrm{e}}{T_0} \tag{2.46}$$

Devices with no gain (e.g., attenuators) have a noise figure equal to their attenuation L (absolute value, not in dB) when their physical temperature equals T_0. More generally, for an attenuator at a physical temperature, the noise temperature is $T_e = (L - 1)T_{phys}$, thus giving a noise factor of

$$F = 1 + \frac{(L-1)T_\mathrm{phys}}{T_0} \tag{2.47}$$

If several devices are cascaded, the total noise factor can be found with Frii's Formula:

$$F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \frac{F_4 - 1}{G_1 G_2 G_3} + \cdots + \frac{F_n - 1}{G_1 G_2 G_3 \cdots G_{n-1}}, \tag{2.48}$$

where F_n is the noise factor for the n-th device and G_n is the power gain of the n-th device. As an example, let's assume that an amplifier at room temperature with 10dB of gain which has only a matched resistor at its output and output. The noise at the input of the amplifier must be -174dBm/Hz. If the amplifier is known to have a 3dB NF, the internal noise source adds an equal noise to the input noise before amplification. Then 10dB of gain increases the noise by 10dB. Therefore, the noise at the output of the amplifier is 13dB higher than at the input, or (-174dBm/Hz + 10dB gain +3dB NF) = -161dBm/Hz.

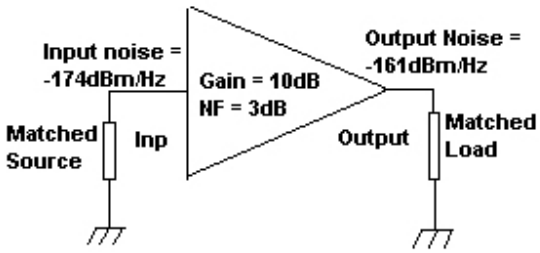

Figure 2.9 Amplifier at Room Temperature

The noise contribution of the amplifier's noise source is fixed and does not change with input signal. Therefore, when more noise is present at the amplifier input, the contribution of the internal noise source is less significant in comparison. When the noise into an amplifier is higher than kTB (-174dBm/Hz), the amplifier 's Noise Figure plays a smaller role in the amplifier's noise contribution. The Noise Figure (NF) of a device is only calculated with the input noise level at kTB. The Noise Temperature at the output of an amplifier is the sum of the Noise Temperature of the source and the Noise Temperature of the amplifier itself multiplied by the Power Gain of the amplifier.

$$T_{out} = G * (T_{ampl} + T_{source}) \qquad (2.49)$$

Where, T_{out} = Noise Temperature at amplifier output in degrees kelvin

G = Power Gain in linear scale not dB.

T_{ampl} = Noise Temperature of amplifier.

T_{source} = Noise Temperature of source.

The Noise Figure of an attenuator is the same as the attenuation in dB. The Noise Figure of an attenuator preceding an amplifier is the Noise Figure of the amplifier plus the attenuation of the attenuator in dB.

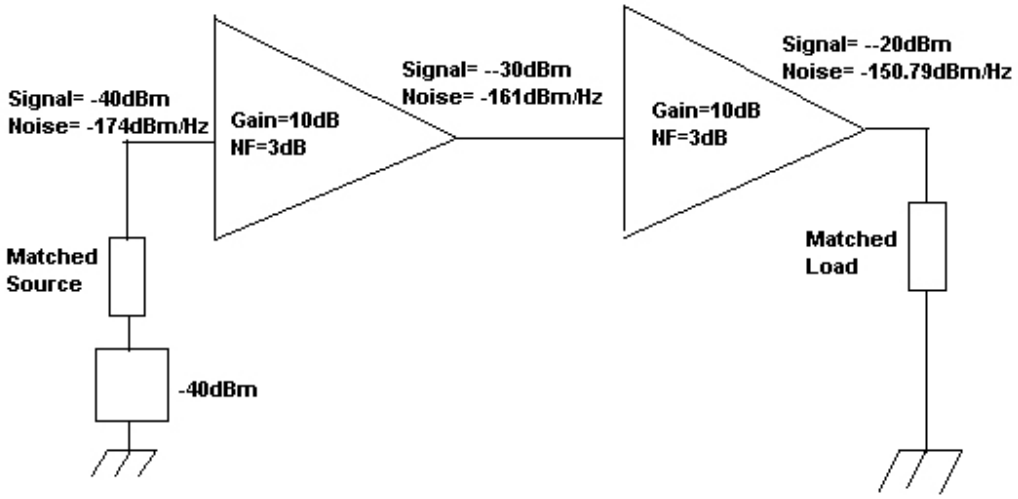

Figure 2.10 Cascaded Amplifier

For above example, both amplifiers has 10dB gain and NF=3dB. The signal goes in at -40dBm with a noise floor at kTB (-174dBm/Hz). The signal at the output of the first amplifier is -30dBm and the noise is (-174dBm/Hz input noise) + (10dB of gain) + (3dB NF) = -161dBm/Hz. 13dB is a power ratio of 20x. So, the noise floor at the second amplifier is 20 times kTB or 20kTB. Next calculate how many kTBs are added by the noise source of the second amplifier (in this case, 1kTB because the NF=3dB). Finally calculate the increase in noise floor at the second amplifier as a ratio and convert to dB. Ratio of (input noise floor) + (added noise) to (input noise floor) is (20kTB+1kTB) / (20kTB)= 20/21. In dB = 10LOG (21/20)=0.21dB. Therefore, the second amplifier only increases the noise floor by 0.21dB even though it has a noise figure of 3dB, simply because the noise floor at its input is significantly higher than kTB. The first amplifier degrades the signal to noise ratio by 3dB, while the second amplifier degrades it only 0.21dB. When amplifiers are cascaded together in order to amplify very weak signals, it is generally the first amplifier in the chain which will have the greatest influence upon the signal to noise ratio because the noise floor is lowest at that point in the chain. Determining the total Noise Figure of a chain of amplifiers (or other devices):

NFactor_total = NFact1 + (NFact2-1)/G1 + (NFact3-1)/(G1*G2) +(NFact3-1)/(G1*G2*….Gn-1) (2.50)

Where, NFactor = Noise factor of each stage (Linear not in dB).

Noise Figure[dB] = 10*LOG(NFact) (2.51)

G = Gain of each stage as a ratio, not dB (for example 4x, not 6dB)

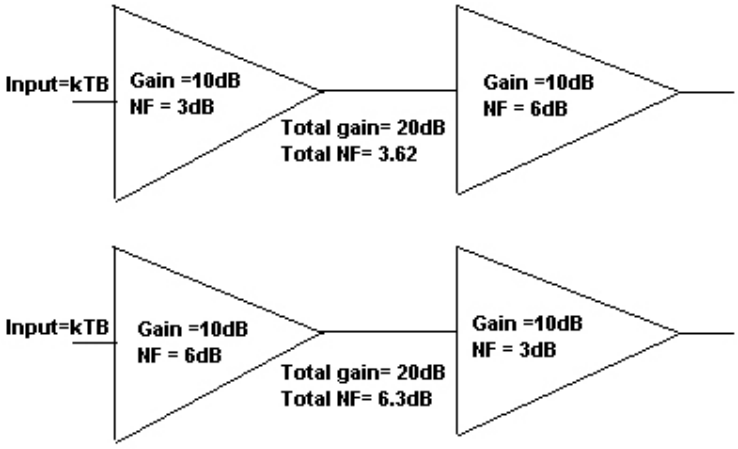

Figure 2.11 Chain Of Amplifiers

The first amplifier in a chain has the most significant effect on the total noise figure than any other amplifier in the chain. The lower noise figure amplifier should usually go first in a line of amplifiers (assuming all else is equal). If we have two amplifiers with equal gain, but with different noise figures. Assume 10dB gain in each amplifier. One amp is NF = 3dB and the other 6dB. When the 3dB NF amplifier is first in cascade, the total noise figure, the total NF is 3.62dB. When the 6dB amplifier is first, the total NF is 6.3dB. This also applies to gain.

- If two amplifiers have the same noise figure but different gains, the higher gain amplifier should precede the lower gain amplifier to achieve the best overall noise figure.

The overall Noise Factor of an infinite number of identical cascaded amplifiers is:

NFactor_total = 1 + M (2.52)

Where, M = (F-1) / [1 - (1/G)] (2.53)

Where, F is Noise_Factor(linear) of each stage, and G is Gain(linear) of each stage.

NoiseFigure_total[dB] = 10*LOG (NFactor_total(linear)) (2.54)

All the devices which process a signal contribute noise such as Amplifiers, Mixers, transistors, diodes, etc all have noise figures.

- For example, RF attenuators have a noise figure floor equal to their attenuation value. A 10dB pad has a 10dB NF. If a signal enters in a pad and the noise floor is

at -174 dBm/Hz the signal is attenuated by 10dB while the noise floor remains constantly (it cannot get any lower than -174 dBm/Hz at room temperature). Therefore the signal to noise ratio through the pad is degraded by 10dB. Like amplifiers, if the noise floor is above kTB, the signal to noise ratio degradation of the pad will be less than its noise figure.

- The Radiation Resistance of the antenna does not convert power to heat, and so is not a source of thermal noise.
- The load impedance of the input of the receiver does not contribute directly to received noise. Therefore, it is indeed possible, and even common, for a receiver to have a Noise Factor of less than 2x (or equivalently, a Noise Figure of less than 3dB).

2.9.2 Noise Figure Measurement Techiques: Practical Approach

There are atleast three ways to measure the noise figure of a device. These are:

1. Using Noise Figure Meter
2. Gain Method,
3. Y Factor Method

2.9.2.1 Noise Figure Measurement using Noise Figure Meter:

The equipment connections are as shown in the figure 2.9 below. A mixer may be necessary if want to convert the RF frequencies to desired IF frequencies. In case the frequency mixer is not needed, connect the IF OUT cable to the Noise Source, and calibrate.

Excess Noise Ratio (ENR): ENR is frequently used to denote the noise that a calibrated noise source delivers to a DUT. The ENR needs to be entered into the Noise Figure Analyzer corresponding to the frequency of measurement. Normally, the ENR table is prominently displayed on the Noise Source.

$$ENR = [T_H - T_C]/T_O \qquad (2.55)$$

Where T_H is the hot temperature (Corresponds to Noise source ON) and T_C is the Cold temperature (Corresponds to Noise source OFF).

b. Calibration procedure: First the equipment needs to be calibrated. The calibration procedure normally involves inputting the ENR (Excess Noise Ratio) as given on the Noise Source in to the Noise Figure Analyzer at the desired frequency range.

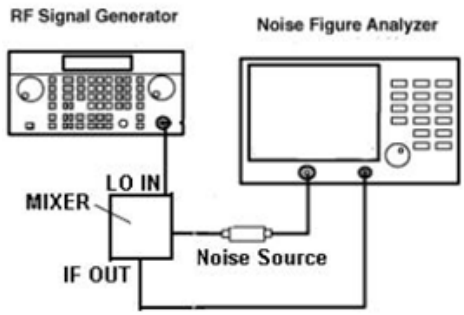

Figure 2.12 Measurement of Noise Figure and Gain

The following figure 2.13 shows the diagram with required connections with DUT. Once the test equipment is calibrated for Noise Figure, (and Gain, as it is normally measured along with the Noise Figure) by simply connecting the DUT into the calibrated set up as below will display the Gain and Noise Figure of the DUT. It is possible that to use any connector adapters or attenuators during the measurement process to ensure that the measured values are within the range of the Noise Figure Analyzer. In such case include the adapters or attenuators during the calibration process itself (not shown in the figure below).

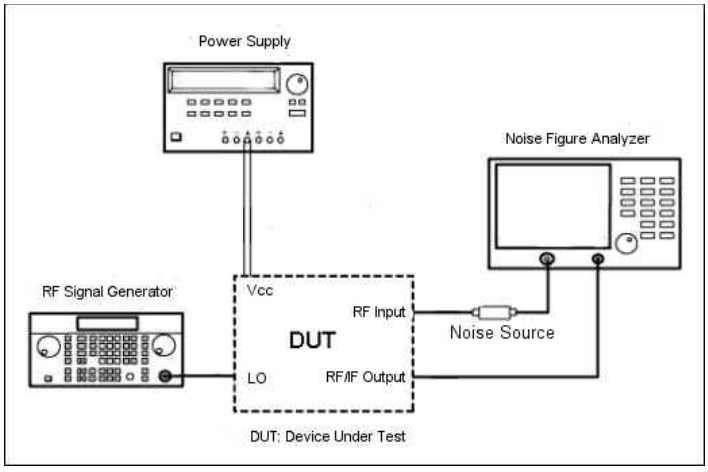

Figure 2.13 Diagram with required connections with DUT

The following are the factors that need to be considered when choosing equipment for Noise Figure measurement:

1. Expected Noise Figure: NF Analyzer is suitable to measure Noise Figure when the value is small (say less than 10dB typical). If intend to measure very high value or very low value (say less than 0.05dB), consult the manufacturer for suitability of the equipment.

2. Due to the limitations of the DUT/NF Analyzer, use external mixer for measuring NF. In such an event, ensure that the NF Analyzer supports external/internal mixer capabilities.
3. DUT Connectors: Some times, the DUT may have different connectors, such as waveguide. If so, procure appropriate waveguide to co-axial adapter. Most of the NF Analyzers use only co-axial connector for interconnectivity.
4. Gain Measurement: Normally, measure gain along with NF. Ensure that the NF Analyzer is capable of measuring expected range of DUT Gain.
5. For frequency conversion requires external frequency generator, and a mixer. Most measurement (such as Up/Down Converters) require external frequency generator.

2.9.2.2 Gain Method

The Noise figure measurement using gain method is based on the following formulas:

$F = [GN_{in} + N_{dut]}/[GN_{in}]$

$=1+ N_{dut]}/[GN_{in}]$ (2.56)

Or $10\log F = 10\log N_{dut} - 10 \log N_{in}$ (2.57)

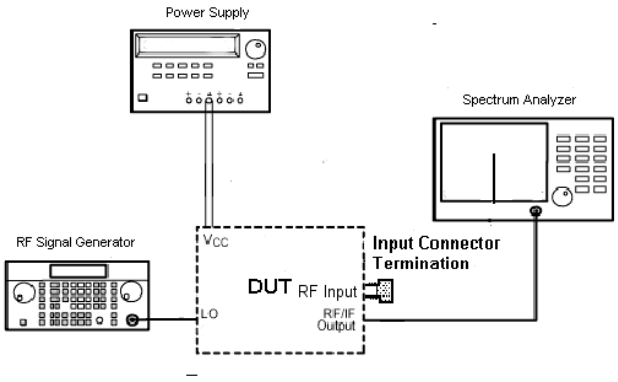

Figure 2.14 Diagram for measurement of NF using Gain method

The advantages of using this method are as follows:

1. This method is very useful for measuring very high Noise Figure (or the order of 10dB or more).
2. The method is more intuitive, and useful for experimentation.

3. A spectrum analyzer can also be used to make other measurements such as amplifier harmonics, gain, etc. where as a NF analyzer is specifically made for NF measurements.
4. Gain method is recommended when making measurements at low frequencies, typically less than 10MHz.

The Disadvantages of using this method are as follows:

1. The spectrum analyzer should be able to provide very good resolution bandwidth, and noise floor, typically of the order of –130dBm. A spectrum analyzer becomes very expensive when need to measure very low signal levels such as –130dBm at high resolution bandwidths (typically few Hertz).
2. This method requires that the Gain of the DUT is known already. Also, the accuracy of Noise Figure measured depends directly on the accuracy of the measured Gain.

2.9.2.3 Y-Factor Method of Noise Figure Measurement

The Y-Factor is the ratio of Hot and Cold noise powers (in watts) and is defined as:

$$Y = N_h/N_c \quad (2.58)$$

If the Noise source is at room temperature, then $N_c = N_o$ and the equation becomes,

$$Y = N_f/N_o \quad (2.59)$$

Note that the Y factor method is a relative method and does not depend on the rest of the equipment. All you need is to measure the power levels accurately while the noise source is OFF and ON.The Noise Figure is related to the Y factor as below:

$$F = ENR/[Y-1] \quad (2.60)$$

All the above parameters are in linear units. Normally, the ENR provided on the noise source is in decibels. This needs to be converted to linear units for computing the noise figure.

The Advantages of this method are as follows:

1. The equipment required is less. The need of a noise source and power meter to measure the power levels with noise source ON and OFF. Of course, a need of a mechanism to turn the noise source ON and OFF.
2. The method can be used to measure noise figure over a wide frequency range.

The Disadvantages of using this method are as follows:

1. Due to the limitation of noise source, if the DUT noise figure is very high, the results may not be very accurate.
2. The other equipment needs to be stable to get repeatable measurements.

2.10 NOISE TEMPERATURE

Noise temperature is a temperature (in kelvin) assigned to a component such that the noise power delivered by the noisy component to a noiseless matched resistor is given by

$$P_{RL} = kT_sB_n \qquad (2.61)$$

in Watts, where:

- K = Boltzmann Constant (1.381×10^{-23} J/K, joules per kelvin)
- T_s = noise temperature (K)
- B_n = noise bandwidth (Hz)

Engineers often model noisy components as an ideal component in series with a noisy resistor. The source resistor is often assumed to be at room temperature, conventionally taken as 290 K (17 °C, 62 °F).[10]

2.10.1 Applications

A communications system is typically made up of a transmitter, a communications channel, and a receiver. The communications channel may consist of any one or a combination of many different physical media (air, coaxial cable, printed wiring board traces…). The important thing to note is that no matter what physical media the channel consists of, the transmitted signal will be randomly corrupted by a number of different processes. The most common form of signal degradation is called additive noise.[10]

The additive noise in a receiving system can be of thermal origin (thermal noise) or can be from other noise-generating processes. Most of these other processes generate noise whose spectrum and probability distributions are similar to thermal noise. Because of these similarities, the contributions of all noise sources can be lumped together and regarded as thermal noise. The noise power generated by all these sources (P_n) can be described by assigning to the noise a noise temperature (T_n) defined as:[7]

$$T_n = P_n/(kB_n) \qquad (2.62)$$

In a wireless communications receiver, T_n would equal the sum of two noise temperatures:

$$T_n = (T_{ant} + T_{sys}) \quad (2.63)$$

T_{ant} is the antenna noise temperature and determines the noise power seen at the output of the antenna. The physical temperature of the antenna has no affect on T_{ant} and T_{sys} is the noise temperature of the receiver circuitry and is representative of the noise generated by the non-ideal components inside the receiver. An important application of noise temperature is its use in the determination of a component's noise factor. The noise factor quantifies the noise power that the component adds to the system when its input noise temperature is given by:

$$F = \frac{T_0 + T_{sys}}{T_0} \quad (2.64)$$

If there are multiple noisy components in cascade, the noise temperature of the cascade will be limited by the noise temperature of the first component in the cascade. The gains of the components at the beginning of the cascade have a large effect on the contribution of later stages to the overall cascade noise temperature. The cascade noise temperature can be found using:

$$T_{cas} = T_1 + \frac{T_2}{G_{p,1}} + \frac{T_3}{G_{p,1}G_{p,2}} + \ldots + \frac{T_n}{G_{p,1}G_{p,2}G_{p,3}\ldots G_{p,n-1}} \quad (2.65)$$

Where,

- T_{cas}= cascade noise temperature
- T_1= noise temperature of the first component in the cascade
- T_2= noise temperature of the second component in the cascade
- T_n= noise temperature of the nth component in the cascade
- $G_{p,1}$= linear gain of the first component in the cascade
- $G_{p,2}$= linear gain of the second component in the cascade
- $G_{p,3}$= linear gain of the third component in the cascade
- $G_{p,n-1}$ = linear gain of the (n-1) component in the cascade

2.10.2 Measuring Noise Temperature

The direct measurement of a component's noise temperature is a difficult process. Suppose that the noise temperature of a low noise amplifier (LNA) is measured by connecting a noise source to the LNA with a piece of transmission line. From the cascade

noise temperature it can be seen that the noise temperature of the transmission line (T_1) has the potential of being the largest contributor to the output measurement (especially when you consider that LNA's can have noise temperatures of only a few Kelvin). To accurately measure the noise temperature of the LNA the noise from the input coaxial cable needs to be accurately known. This is difficult because poor surface finishes and reflections in the transmission line make actual noise temperature values higher than those predicted by theoretical analysis. Similar problems arise when trying to measure the noise temperature of an antenna. Since the noise temperature is heavily dependent on the orientation of the antenna, the direction that the antenna was pointed during the test needs to be specified. In receiving systems, the system noise temperature will have three main contributors, the antenna, the transmission line and the receiver circuitry temperature. The antenna noise temperature is considered to be the most difficult to measure because the measurement must be made in the field on an open system. One technique for measuring antenna noise temperature involves using cryogenically cooled loads to calibrate a noise figure meter before measuring the antenna. This provides a direct reference comparison at a noise temperature in the range of very low antenna noise temperatures, so that little extrapolation of the collected data is required.[10]

CHAPTER - 3

LABVIEW

A LabVIEW or Laboratory Virtual Instrument Engineering Workbench is graphical programming language software used for data acquisition and instrument control. The programs that take a long time using conventional programming languages can be completed in hours using LabVIEW. The LabVIEW programs are called virtual instruments (*VIs*) because their appearance and operation imitate actual instruments.

The graphical language is named "G". Originally released for the Apple Macintosh in 1986, LabVIEW is commonly used for data acquisition, instrument control, and industrial automation on a variety of platforms including Microsoft Windows, various flavors of UNIX, Linux, and Mac OS X.

A VI has two main parts as follows:

3.1 FRONT PANEL

It is the interactive user interface of a VI. The front panel can contain knobs, push buttons etc. which are the interactive input controls and graphs, LED's etc. which are indicators. Controls simulate instrument input devices and supply data to block diagram of the VI. Indicators simulate output devices and display the data that block diagram generates. The front panel objects have corresponding terminal on the block diagram.

3.2 BLOCK DIAGRAM

It is the VI's source code, constructed in LabVIEW's graphical programming language, G. The block diagram is the actual executable program. The components of the block diagram are lower level VIs, built in functions, constants and program execution control structures. Wires are used to connect the objects together to indicate the flow of data between them. The data that enter into the front panel controls enter the block diagram through these terminals and after execution the output data flow to indicator terminals where they exit the block diagram, re-enter the front panel and appear in front panel indicators giving us final results.

3.3 DATAFLOW PROGRAMMING

The programming language used in LabVIEW, also referred to as G, is a dataflow programming language. Execution is determined by the structure of a graphical block diagram (the LV-source code) on which the programmer connects different function-nodes by drawing wires. These wires propagate variables and any node can execute as soon as all its input data become available. Since this might be the case for multiple nodes simultaneously, G is inherently capable of parallel execution. Multi-processing and multi-threading hardware is automatically exploited by the built-in scheduler, which multiplexes multiple OS threads over the nodes ready for execution.

3.4 GRAPHICAL PROGRAMMING

LabVIEW ties the creation of user interfaces (called front panels) into the development cycle. LabVIEW programs/subroutines are called virtual instruments (VIs). Each VI has three components: a block diagram, a front panel, and a connector panel. The last is used to represent the VI in the block diagrams of other, calling VIs. Controls and indicators on the front panel allow an operator to input data into or extract data from a running virtual instrument. However, the front panel can also serve as a programmatic interface. Thus a virtual instrument can either be run as a program, with the front panel serving as a user interface, or, when dropped as a node onto the block diagram, the front panel defines the inputs and outputs for the given node through the connector pane. This implies each VI can be easily tested before being embedded as a subroutine into a larger program.

The graphical approach also allows non-programmers to build programs simply by dragging and dropping virtual representations of lab equipment with which they are already familiar. The LabVIEW programming environment, with the included examples and the documentation, makes it simple to create small applications. This is a benefit on one side, but there is also a certain danger of underestimating the expertise needed for good quality "G" programming. For complex algorithms or large-scale code, it is important that the programmer possess an extensive knowledge of the special LabVIEW syntax and the topology of its memory management. The most advanced LabVIEW development systems offer the possibility of building stand-alone applications. Furthermore, it is possible to create distributed applications, which communicate by a

client/server scheme, and are therefore easier to implement due to the inherently parallel nature of *G*-code.

3.5 BENEFITS

The benefit of LabVIEW over other development environments is the extensive support for accessing instrumentation hardware. Drivers and abstraction layers for many different types of instruments and buses are included or are available for inclusion. These present themselves as graphical nodes. The abstraction layers offer standard software interfaces to communicate with hardware devices. The provided driver interfaces save program development time. The sales pitch of National Instruments is, therefore, that even people with limited coding experience can write programs and deploy test solutions in a reduced time frame when compared to more conventional or competing systems. A new hardware driver topology (DAQmxBase), which consists mainly of G-coded components with only a few register calls through NI Measurement Hardware DDK (Driver Development Kit) functions, provides platform independent hardware access to numerous data acquisition and instrumentation devices. The DAQmxBase driver is available for LabVIEW on Windows, Mac OS X and Linux platforms.

In terms of performance, LabVIEW includes a compiler that produces native code for the CPU platform. The graphical code is translated into executable machine code by interpreting the syntax and by compilation. The LabVIEW syntax is strictly enforced during the editing process and compiled into the executable machine code when requested to run or upon saving. In the latter case, the executable and the source code are merged into a single file. The executable runs with the help of the LabVIEW run-time engine, which contains some precompiled code to perform common tasks that are defined by the G language. The run-time engine reduces compile time and also provides a consistent interface to various operating systems, graphic systems, hardware components, etc. The run-time environment makes the code portable across platforms. Generally, LV code can be slower than equivalent compiled C code, although the differences often lie more with program optimization than inherent execution speed.

Many libraries with a large number of functions for data acquisition, signal generation, mathematics, statistics, signal conditioning, analysis, etc., along with numerous graphical interface elements are provided in several LabVIEW package options. The number of

advanced mathematic blocks for functions such as integration, filters, and other specialized capabilities usually associated with data capture from hardware sensors is immense. In addition, LabVIEW includes a text-based programming component called MathScript with additional functionality for signal processing, analysis and mathematics. MathScript can be integrated with graphical programming using "script nodes" and uses a syntax that is generally compatible with MATLAB. The fully object-oriented character of LabVIEW code allows code reuse without modifications: as long as the data types of input and output are consistent, two sub VIs are interchangeable.The LabVIEW Professional Development System allows creating stand-alone executables and the resultant executable can be distributed an unlimited number of times. The run-time engine and its libraries can be provided freely along with the executable.

3.6 RELEASE HISTORY OF LABVIEW

Table 3.1 Release History of labVIEW

Version	Date
LabVIEW 1.0 (for Macintosh)	1986
LabVIEW 2.0	1990
LabVIEW (for Sun & Windows)	1992
LabVIEW (Multiplatform)	1993
LabVIEW 4.0	1997
LabVIEW 5.0	1998
LabVIEW Real-Time	1999
LabVIEW 6i	2000
LabVIEW 7 Express	2003
LabVIEW 8	2005
LabVIEW 8.5	2/19/2008
LabVIEW 8.6	7/24/2008
LabVIEW 2009 (32 and 64-bit)	8/4/2009

3.7 REPOSITORIES AND LIBRARIES

OpenG, as well as LAVA Code Repository (LAVAcr), serve as repositories for a wide range of LabVIEW applications and libraries.VI Package Manager has become the standard package manager for LabVIEW libraries. It is very similar in purpose to Ruby's RubyGems and Perl's CPAN, although it provides a graphical user interface similar to the Synaptic Package Manager. VI Package Manager provides access to a repository of the OpenG (and other) libraries for LabVIEW.

CHAPTER - 4

DIGITAL FILTER

A digital filter is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. This is in contrast to the other major type of electronic filter, the analog filter, which is an electronic circuit operating on continuous-time analog signals. An analog signal may be processed by a digital filter by first being digitized and represented as a sequence of numbers, then manipulated mathematically, and then reconstructed as a new analog signal. In an analog filter, the input signal is "directly" manipulated by the circuit. A digital filter system usually consists of an analog-to-digital converter (to sample the input signal), a microprocessor (often a specialized digital signal processor), and a digital-to-analog converter. Software running on the microprocessor can implement the digital filter by performing the necessary mathematical operations on the numbers received from the ADC. In some high performance applications, an FPGA or ASIC is used instead of a general purpose microprocessor.

Digital filters may be more expensive than an equivalent analog filter due to their increased complexity, but they make practical many designs that are impractical or impossible as analog filters. Since digital filters use a sampling process and discrete-time processing, they experience latency (the difference in time between the input and the response), which is almost irrelevant in analog filters. Digital filters becames an essential element of everyday electronics such as radios, cellphones, and stereo receivers.

4.1 ADVANTAGES OF DIGITAL FILTER

A digital filter is programmable, i.e. its operation is determined by a program stored in the processor's memory. This means the digital filter can easily be changed without affecting the circuitry (hardware). An analog filter can only be changed by redesigning the filter circuit. Digital filters are easily designed, tested and implemented on a general-purpose computer or workstation. The characteristics of analog filter circuits (particularly those containing active components) are subject to drift and are dependent on temperature. Digital filters do not suffer from these problems, and so are extremely stable

with respect both to time and temperature. Unlike their analog counterparts, digital filters can handle low frequency signals accurately. As the speed of DSP technology continues to increase, digital filters are being applied to high frequency signals in the RF (radio frequency) domain, which in the past was the exclusive preserve of analog technology.

Digital filters are very much more versatile in their ability to process signals in a variety of ways; this includes the ability of some types of digital filter to adapt to changes in the characteristics of the signal. Fast DSP processors can handle complex combinations of filters in parallel or cascade (series), making the hardware requirements relatively simple and compact in comparison with the equivalent analog circuitry.

4.2 CHARACTERIZATION OF DIGITAL FILTERS

A digital filter is characterized by its transfer function, or equivalently, its difference equation. Mathematical analysis of the transfer function can describe how it will respond to any input. As such, designing a filter consists of developing specifications appropriate to the problem (for example, a second-order lowpass filter with a specific cut-off frequency), and then producing a transfer function which meets the specifications.The transfer function for a linear, time-invariant, digital filter can be expressed as a transfer function in the Z-domain; if it is causal, then it has the form:

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_N z^{-N}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_M z^{-M}} \qquad (4.1)$$

where the order of the filter is the greater of N or M. This form is for a recursive filter, which typically leads to infinite impulse response behaviour, but if the denominator is unity, then this is the form for a finite impulse response filter.

4.3 ANALYSIS TECHNIQUES

A variety of mathematical techniques may be employed to analyze the behaviour of a given digital filter. Many of these analysis techniques may also be employed in designs, and often form the basis of a filter specification. Typically, one analyzes filters by calculating how the filter will respond to a simple input. One can then extend this information to visualize the filter's response to more complex signals.

4.3.1 Impulse Response

The impulse response, often denoted H(z) or h(n) is a measurement of how a filter will respond to the Kronecker delta function. For example, given a difference equation, one

would set $x(0) = 1$ and $x(n) = 0$ for $n > 0$ and evaluate. In the case of linear time-invariant FIR filters, the impulse response is exactly equal to the sequence of filter coefficients $h(n) = b_n$. In general, the impulse response is a characterization of the filter's behavior. A plot of the impulse response will help to reveal how a filter will respond to a sudden, momentary disturbance.

4.3.2 Difference Equation

In discrete-time systems, the digital filter is often implemented by converting the transfer function to a linear constant-coefficient difference equation (LCCD) via the Z-transform. The discrete frequency-domain transfer function is written as the ratio of two polynomials. For example:

$$H(z) = \frac{(z+1)^2}{(z-\frac{1}{2})(z+\frac{3}{4})} \tag{4.2}$$

This is expanded:

$$H(z) = \frac{z^2+2z+1}{z^2+\frac{1}{4}z-\frac{3}{8}} \tag{4.3}$$

and divided by the highest order of z:

$$H(z) = \frac{1+2z^{-1}+z^{-2}}{1+\frac{1}{4}z^{-1}-\frac{3}{8}z^{-2}} = \frac{Y(z)}{X(z)} \tag{4.4}$$

The coefficients of the denominator, a_k, are the 'feed-backward' coefficients and the coefficients of the numerator are the 'feed-forward' coefficients, b_k. The resultant linear difference equation is:

$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k] \tag{4.5}$$

and finally, by solving for $y[n]$:

$$y[n] = -\frac{1}{4}y[n-1] + \frac{3}{8}y[n-2] + x[n] + 2x[n-1] + x[n-2] \tag{4.6}$$

This equation shows how to compute the next output sample, y[n], in terms of the past outputs, y[n − p], the present input, x[n], and the past inputs, x[n − p]. Applying the filter to an input in this form is equivalent to a Direct Form I or II realization, depending on the exact order of evaluation.

4.4 FILTER DESIGN

The design of digital filters is a deceptively complex topic.[23] Although filters are easily understood and calculated, the practical challenges of their design and implementation are significant and are the subject of much advanced research. There are two categories of digital filter: the recursive filter and the nonrecursive filter. These are often referred to as infinite impulse response (IIR) filters and finite impulse response (FIR) filters, respectively.[23]

4.5 FILTER REALIZATION

After a filter is designed, it must be realized by developing a signal flow diagram that describes the filter in terms of operations on sample sequences.A given transfer function may be realized in many ways. Consider how a simple expression such as $ax + bx + c$ could be evaluated – one could also compute the equivalent $x(a + b) + c$. In the same way, all realizations may be seen as "factorizations" of the same transfer function, but different realizations will have different numerical properties. Specifically, some realizations are more efficient in terms of the number of operations or storage elements required for their implementation, and others provide advantages such as improved numerical stability and reduced round-off error. Some structures are more optimal for fixed-point arithmetic and others may be more optimal for floating-point arithmetic.

4.5.1 Direct Form I

A straightforward approach for IIR filter realization is Direct Form I, where the difference equation is evaluated directly. This form is practical for small filters, but may be inefficient and impractical (numerically unstable) for complex designs.[17] In general, this form requires 2N delay elements (for both input and output signals) for a filter of order N.

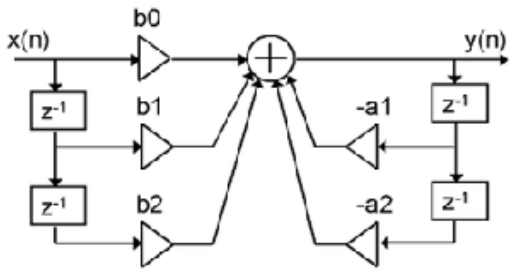

Figure 4.1: Realization of Direct Form I

4.5.2 Direct Form II

The alternate Direct Form II only needs N delay units, where N is the order of the filter – potentially half as much as Direct Form I. This structure is obtained by reversing the order of the numerator and denominator sections of Direct Form I, since they are in fact two linear systems, and the commutativity property applies. There are two columns of delays (z − 1) that tap off the center net, and these can be combined since they are redundant, yielding the implementation as shown in figure. The disadvantage is that Direct Form II increases the possibility of arithmetic overflow for filters of high Q or resonance.[19] As Q increases, the round-off noise of both direct form topologies increases without bounds. This is because, conceptually, the signal is first passed through an all-pole filter (which normally boosts gain at the resonant frequencies) before the result of that is saturated, then passed through an all-zero filter (which often attenuates much of what the all-pole half amplifies).

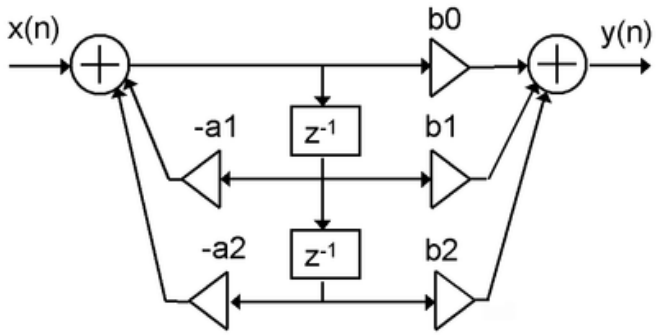

Figure 4.2: Realization of Direct Form II

4.5.3 Cascaded Second Order Sections

A common strategy is to realize a higher-order (greater than 2) digital filter as a cascaded series of second-order "biquadratric" (or "biquad") sections. Cascading direct form II sections result in N delay elements for filter order of N. Cascading direct form I sections result in N+2 delay elements since the delay elements of the input of any section (except the first section) are a redundant with the delay elements of the output of the preceding section. Other forms include:

- Direct Form I and II transpose
- Series/cascade
- Parallel

- Ladder form
- Lattice form
- Coupled normal form
- Multifeedback
- Analog-inspired forms such as Sallen-key and state variable filters
- Systolic arrays

4.6 COMPARISON OF ANALOG AND DIGITAL FILTERS

Digital filters are not subject to the component non-linearities that greatly complicate the design of analog filters. Analog filters consist of imperfect electronic components, whose values are specified to a limit tolerance (e.g. resistor values often have a tolerance of +/- 5%) and which may also change with temperature and drift with time. As the order of an analog filter increases, and thus its component count, the effect of variable component errors is greatly magnified. In digital filters, the coefficient values are stored in computer memory, making them far more stable and predictable.[17] Because the coefficients of digital filters are definite, they can be used to achieve much more complex and selective designs – specifically with digital filters, one can achieve a lower passband ripple, faster transition, and higher stopband attenuation than is practical with analog filters. Even if the design could be achieved using analog filters, the engineering cost of designing an equivalent digital filter would likely be much lower. Furthermore, one can readily modify the coefficients of a digital filter to make an adaptive filter or a user-controllable parametric filter. While these techniques are possible in an analog filter, they are again considerably more difficult. Digital filters can be used in the design of finite impulse response filters. Analog filters do not have the same capability, because finite impulse response filters require delay elements. Digital filters rely less on analog circuitry, potentially allowing for a better signal-to-noise ratio. A digital filter will introduce noise to a signal during analog low pass filtering, analog to digital conversion, digital to analog conversion and may introduce digital noise due to quantization.

With analog filters, every component is a source of thermal noise (such as Johnson noise), so as the filter complexity grows, so does the noise. However, digital filters do introduce a higher fundamental latency to the system. In an analog filter, latency is often negligible; it is the time for an electrical signal to propagate through the

filter circuit. In digital filters, latency is a function of the number of delay elements in the system. Digital filters also tend to be more limited in bandwidth than analog filters. High bandwidth digital filters require expensive ADC/DACs and fast computer hardware for processing. In very simple cases, it is more cost effective to use an analog filter.

4.7 TYPES OF DIGITAL FILTERS

Many digital filters are based on the Fast Fourier transform, a mathematical algorithm that quickly extracts the frequency spectrum of a signal, allowing the spectrum to be manipulated (such as to create band-pass filters) before converting the modified spectrum back into a time-series signal. Another form of a digital filter is that of a state-space model.

4.7.1 Bessel Filter

In electronics and signal processing, a Bessel filter is a type of linear filter with a maximally flat group delay (maximally linear phase response). Bessel filters are often used in audio crossover systems. Analog Bessel filters are characterized by almost constant group delay across the entire passband, thus preserving the wave shape of filtered signals in the passband. The filter's name is a reference to Friedrich Bessel, a German mathematician (1784–1846), who developed the mathematical theory on which the filter is based.

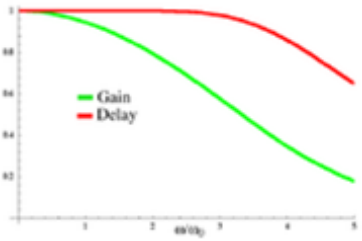

Figure 4.3 Plot of Gain And Group Delay for Bessel Filter

The transition from the pass band to the stop band is much slower than for other filters, but the group delay is practically constant in the passband. The Bessel filter maximizes the flatness of the group delay curve at zero frequency.

A Bessel low-pass filter is characterized by its transfer function:[17]

$$H(s) = \frac{\theta_n(0)}{\theta_n(s/\omega_0)} \qquad (4.7)$$

where $\theta_n(s)$ is a reverse Bessel polynomials from which the filter gets its name and ω_0 is a frequency chosen to give the desired cut-off frequency. The filter has a low-frequency group delay of $1/\omega_0$.

The reverse Bessel polynomials are given by:[17]

$$P(s) = \sum_{k=0}^{N} a_k s^k \tag{4.8}$$

$$a_k = \frac{(2N-k)!}{2^{N-k}k!(N-k)!} \quad k = 0, 1, ..., N \tag{4.9}$$

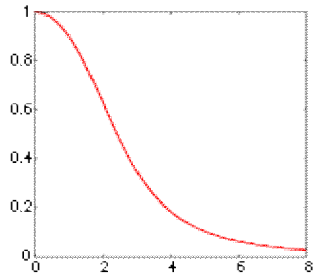

Figure 4.4 Gain Plot of Third Order Bessel Filter

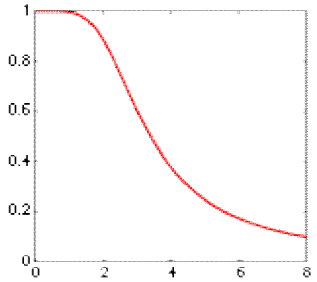

Figure 4.5 Group Delay of Third Order Bessel Filter

The transfer function for a third-order (three-pole) Bessel low-pass filter, normalized to have unit group delay, is

$$H(s) = \frac{15}{s^3 + 6s^2 + 15s + 15} \tag{4.10}$$

The roots of the denominator polynomial, the filter's poles, include a real pole at s = -2.3222, and a complex-conjugate pair of poles plotted above. The numerator 15 is chosen to give unity gain at DC (at s = 0).

The gain is given by:

$$G(\omega) = |H(j\omega)| = \frac{15}{\sqrt{\omega^6 + 6\omega^4 + 45\omega^2 + 225}} \tag{4.11}$$

And, the phase is given by:

$$\phi(\omega) = -\arg(H(j\omega)) = -\arctan\left(\frac{15\omega - \omega^3}{15 - 6\omega^2}\right) \tag{4.12}$$

The group delay is

$$D(\omega) = -\frac{d\phi}{d\omega} = \frac{6\omega^4 + 45\omega^2 + 225}{\omega^6 + 6\omega^4 + 45\omega^2 + 225} \tag{4.13}$$

The Taylor series expansion of the group delay is

$$D(\omega) = 1 - \frac{\omega^6}{225} + \frac{\omega^8}{1125} + \cdots \tag{4.14}$$

Note that the two terms in ω^2 and ω^4 are zero, resulting in a very flat group delay at ω=0. This is the greatest number of terms that can be set to zero, since there are a total of four coefficients in the third order Bessel polynomial, requiring four equations in order to be defined. One equation specifies that the gain be unity at ω = 0 and a second specifies that the gain be zero , leaving two equations to specify two terms in the series expansion to be zero. This is a general property of the group delay for a Bessel filter of order n. The first *n-1* terms in the series expansion of the group delay will be zero, thus maximizing the flatness of the group delay at ω = 0.

4.7.2 Butterworth Filter

The Butterworth filter is one type of signal processing filter design. It is designed to have a frequency response which is as flat as mathematically possible in the passband. Another name for it is maximally flat magnitude filter**.** The frequency response of the Butterworth filter is maximally flat (has no ripples) in the passband, and rolls off towards zero in the stopband.[18] When viewed on a logarithmic Bode plot, the response slopes off linearly towards negative infinity. For a first-order filter, the response rolls off at −6 dB per octave (−20 dB per decade) (all first-order lowpass filters have the same normalized frequency response). For a second-order lowpass filter, the response ultimately decreases at −12 dB per octave, a third-order at −18 dB, and so on. Butterworth filters have a monotonically changing magnitude function with ω, unlike other filter types that have non-monotonic ripple in the passband and/or the stopband. Compared with a Chebyshev Type I/Type II filter or an elliptic filter, the Butterworth filter has a slower roll-off, and thus will require a higher order to implement a particular stopband specification.

However, Butterworth filters have a more linear phase response in the passband than the Chebyshev Type I/Type II and elliptic filters.

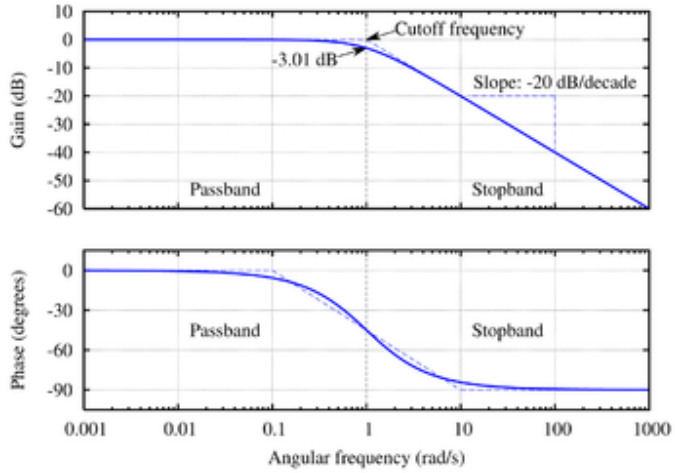

Figure 4.6 Bode plot of First-Order Butterworth Low-Pass Filter

Like all filters, the typical prototype is the low-pass filter, which can be modified into a high-pass filter, or placed in series with others to form band-pass and band-stop filters, and higher order versions. The gain G(ω) of an n-order Butterworth low pass filter is given in terms of the transfer function H(s) as:

$$G^2(\omega) = |H(j\omega)|^2 = \frac{{G_0}^2}{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}} \qquad (4.15)$$

- n = order of filter
- ω_c = cutoff frequency (approximately the -3dB frequency)
- G_0 is the DC gain (gain at zero frequency)

It can be seen that as n approaches infinity, the gain becomes a rectangle function and frequencies below ω_c will be passed with gain G_0, while frequencies above ω_c will be suppressed. For smaller values of n, the cutoff will be less sharp. To determine the transfer function H(s) where $s = \sigma + j\omega$. Since H(s)H(-s) evaluated at $s = j\omega$ is simply equal to $|H(j\omega)|^2$, it follows that:

$$H(s)H(-s) = \frac{{G_0}^2}{1 + \left(\frac{-s^2}{\omega_c^2}\right)^n} \qquad (4.16)$$

The poles of this expression occur on a circle of radius ω_c at equally spaced points. The transfer function itself will be specified by just the poles in the negative real half-plane of *s*. The k-th pole is specified by:

$$-\frac{s_k^2}{\omega_c^2} = (-1)^{\frac{1}{n}} = e^{\frac{j(2k-1)\pi}{n}} \qquad \mathrm{k} = 1, 2, 3, \ldots, \mathrm{n} \tag{4.17}$$

and hence,

$$s_k = \omega_c e^{\frac{j(2k+n-1)\pi}{2n}} \qquad \mathrm{k} = 1, 2, 3, \ldots, \mathrm{n} \tag{4.18}$$

The transfer function may be written in terms of these poles as:

$$H(s) = \frac{G_0}{\prod_{k=1}^{n}(s - s_k)/\omega_c} \tag{4.19}$$

The Butterworth polynomials may be written in complex form as above, but are usually written with real coefficients by multiplying pole pairs which are complex conjugates, such as s_1 and s_n. The polynomials are normalized by setting $\omega_c = 1$. The normalized Butterworth polynomials then have the general form:

$$B_n(s) = \prod_{k=1}^{\frac{n}{2}} \left[s^2 - 2s\cos\left(\frac{2k+n-1}{2n}\pi\right) + 1 \right]; \text{ for n even} \tag{4.20}$$

$$B_n(s) = (s+1)\prod_{k=1}^{\frac{n-1}{2}} \left[s^2 - 2s\cos\left(\frac{2k+n-1}{2n}\pi\right) + 1 \right]; \text{ for n odd} \tag{4.21}$$

Assuming $\omega_c = 1$ and $G_0 = 1$, the derivative of the gain with respect to frequency can be obtained as:

$$\frac{dG}{d\omega} = -nG^3\omega^{2n-1} \tag{4.22}$$

which is monotonically decreasing for all ω since the gain *G* is always positive. The gain function of the Butterworth filter therefore has no ripple. Furthermore, the series expansion of the gain is given by:

$$G(\omega) = 1 - \frac{1}{2}\omega^{2n} + \frac{3}{8}\omega^{4n} + \ldots \tag{4.23}$$

In other words, all derivatives of the gain up to but not including the 2n-th derivative are zero, resulting in "maximal flatness". If the requirement to be monotonic is limited to the passband only and ripples are allowed in the stopband, then it is possible to design a filter of the same order that is flatter in the passband than the "maximally flat" Butterworth. Such a filter is the inverse Chebyshev filter.

4.7.2.1 Filter Design

There are a number of different filter topologies available to implement a linear analogue filter. These circuits differ only in the values of the components, but not in their connections.

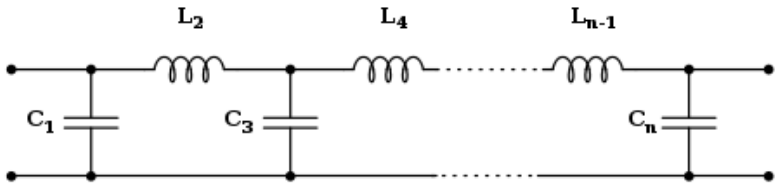

Figure 4.7 Cauer Topology

The Cauer topology uses passive components (shunt capacitors and series inductors) to implement a linear analog filter. The Butterworth filter having a given transfer function can be realized using a Cauer 1-form. The k^{th} element is given by:

$$C_k = 2\sin\left[\frac{(2k-1)}{2n}\pi\right]; \text{ k = odd} \quad (4.23)$$

$$L_k = 2\sin\left[\frac{(2k-1)}{2n}\pi\right]; \text{ k = even} \quad (4.24)$$

The filter may start with a series inductor if desired, in which case the L_k are k odd and the C_k are k even.

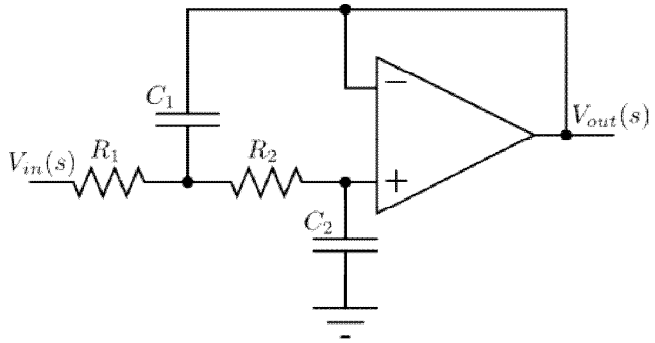

Figure 4.8 Sallen-Key Topology

The Sallen-Key topology uses active and passive components (noninverting buffers, usually op amps, resistors, and capacitors) to implement a linear analog filter. Each Sallen-Key stage implements a conjugate pair of poles; the overall filter is implemented by cascading all stages in series. If there is a real pole (in the case where *n* is odd), this must be implemented separately, usually as an RC circuit, and cascaded with the active

stages. Digital implementations of Butterworth filters often use bilinear transform or matched z-transform to discretize an analog filter. For higher orders, they are sensitive to quantization errors. For this reason, they are often calculated as cascaded biquad sections and a cascaded first order filter, for odd orders.The Butterworth filter rolls off more slowly around the cutoff frequency than the others, but shows no ripples.

4.7.3 Elliptic Filter

An elliptic filter (also known as a Cauer filter, named after Wilhelm Cauer) is a signal processing filter with equalized ripple (equiripple) behavior in both the passband and the stopband. The amount of ripple in each band is independently adjustable, and no other filter of equal order can have a faster transition in gain between the passband and the stopband, for the given values of ripple (whether the ripple is equalized or not). Alternatively, one may give up the ability to independently adjust the passband and stopband ripple, and instead design a filter which is maximally insensitive to component variations. As the ripple in the stopband approaches zero, the filter becomes a type I Chebyshev filter. As the ripple in the passband approaches zero, the filter becomes a type II Chebyshev filter and finally, as both ripple values approach zero, the filter becomes a Butterworth filter.

The gain of a lowpass elliptic filter as a function of angular frequency ω is given by:

$$G_n(\omega) = \frac{1}{\sqrt{1 + \epsilon^2 R_n^2(\xi, \omega/\omega_0)}} \qquad (4.25)$$

where R_n is the *n*th-order elliptic rational function (sometimes known as a Chebyshev rational function) and ω_0 is the cutoff frequency, ε is the ripple factor, ξ is the selectivity factor The value of the ripple factor specifies the passband ripple, while the combination of the ripple factor and the selectivity factor specify the stopband ripple. Its properties are as follows:

- In the passband, the elliptic rational function varies between zero and unity. The passband of the gain therefore will vary.
- In the stopband, the elliptic rational function varies between infinity and the discrimination factor which is defined as:

$$L_n = R_n(\xi, \xi) \qquad (4.26)$$

- In the limit of $\xi \to \infty$ the elliptic rational function becomes a Chebyshev polynomial, and therefore the filter becomes a Chebyshev type I filter, with ripple factor ε.

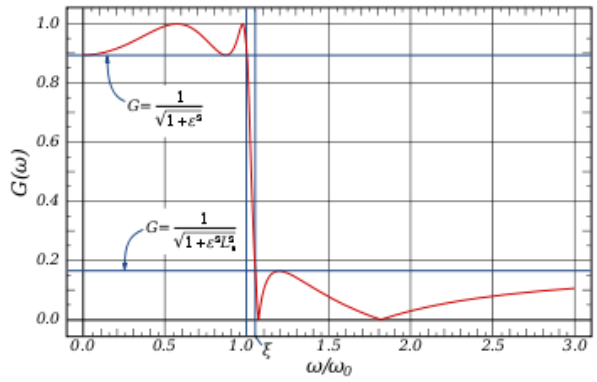

Figure 4.9 Frequency Response of Fourth-Order Elliptic Low-Pass Filter with ε=0.5 and ξ=1.05.

Elliptic filters are generally specified by requiring a particular value for the passband ripple, stopband ripple and the sharpness of the cutoff. This will generally specify a minimum value of the filter order which must be used. Another design consideration is the sensitivity of the gain function to the values of the electronic components used to build the filter. This sensitivity is inversely proportional to the quality factor (Q-factor) of the poles of the transfer function of the filter. The Q-factor of a pole is defined as:

$$Q = -\frac{|s_{pm}|}{2\mathrm{Re}(s_{pm})} = -\frac{1}{2\cos(\arg(s_{pm}))} \qquad (4.27)$$

and is a measure of the influence of the pole on the gain function. For an elliptic filter, it happens that, for a given order, there exists a relationship between the ripple factor and selectivity factor which simultaneously minimizes the Q-factor of all poles in the transfer function:

$$\epsilon_{Qmin} = \frac{1}{\sqrt{L_n(\xi)}} \qquad (4.28)$$

This results in a filter which is maximally insensitive to component variations, but the ability to independently specify the passband and stopband ripples will be lost. For such filters, as the order increases, the ripple in both bands will decrease and the rate of cutoff will increase. If one decides to use a minimum-Q elliptic filter in order to achieve a particular minimum ripple in the filter bands along with a particular rate of cutoff, the

order needed will generally be greater than the order one would otherwise need without the minimum-Q restriction. They will not be evenly spaced and there will be zeroes on the ω axis, unlike the Butterworth filter, whose poles are also arranged in a circle.

4.7.4 Chebyshev Filter

Chebyshev filters are analog or digital filters having a steeper roll-off and more passband ripple (type I) or stopband ripple (type II) than Butterworth filters. Chebyshev filters have the property that they minimize the error between the idealized filter characteristic and the actual over the range of the filter, but with ripples in the passband. This type of filter is named in honor of Pafnuty Chebyshev because their mathematical characteristics are derived from Chebyshev polynomials. Because of the passband ripple inherent in Chebyshev filters, filters which have a smoother response in the passband but a more irregular response in the stopband are preferred for some applications**.**

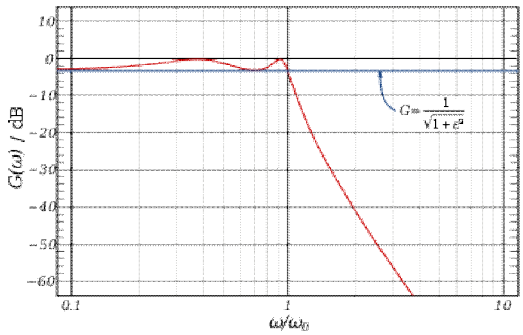

Figure 4.10 Frequency Response of Chebyshev Low Pass Filter

The gain (or amplitude) response as a function of angular frequency ω of the *n*th order low pass filter is

$$G_n(\omega) = |H_n(j\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 T_n^2\left(\frac{\omega}{\omega_0}\right)}} \qquad (4.29)$$

where μ is the ripple factor, ω_0 is the cutoff frequency and T_n is a Chebyshev polynomial of the *n*th order.

The passband exhibits equiripple behavior, with the ripple determined by the ripple factor ε. In the passband, the Chebyshev polynomial alternates between 0 and 1 so the filter gain will alternate between maxima at $G = 1$ and minima at $G = 1/\sqrt{1+\varepsilon^2}$. At the cutoff frequency ω_0 the gain again has some value but continues to drop into the stop band as

the frequency increases. The order of a Chebyshev filter is equal to the number of reactive components needed to realize the filter using analog electronics.

The ripple is often given in dB:

$$\text{Ripple in dB} = 20\log_{10}\frac{1}{\sqrt{1+\varepsilon^2}} \tag{4.30}$$

An even steeper roll-off can be obtained if allowed for ripple in the stop band, by allowing zeroes on the jω-axis in the complex plane. This will however result in less suppression in the stop band. The result is called an elliptic filter, also known as Cauer filter. The above expression yields the poles of the gain G. For each complex pole, there is another which is the complex conjugate, and for each conjugate pair there are two more that are the negatives of the pair. The transfer function must be stable, so that its poles will be those of the gain that have negative real parts and therefore lie in the left half plane of complex frequency space. The transfer function is then given by:

$$H(s) = \frac{1}{2^{n-1}\varepsilon}\prod_{m=1}^{n}\frac{1}{(s-s_{pm}^{-})} \tag{4.31}$$

There are only those poles with a negative sign in front of the real term in the above equation for the poles.

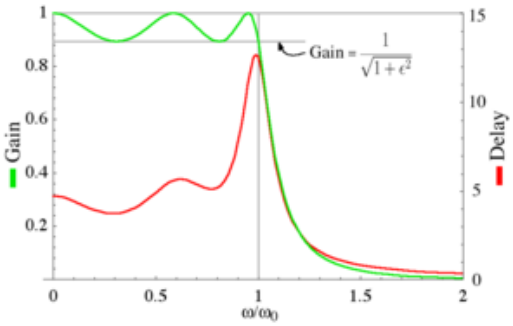

Figure 4.11 Gain And Group Delay of Chebyshev Filter

The group delay is defined as the derivative of the phase with respect to angular frequency and is a measure of the distortion in the signal introduced by phase differences for different frequencies.

$$\tau_g = -\frac{d}{d\omega}\arg(H(j\omega)) \tag{4.32}$$

The gain and the group delay for a fifth order type I Chebyshev filter with ε=0.5 are plotted in the graph. It can be seen that there are ripples in the gain and the group delay in the passband but not in the stop band.

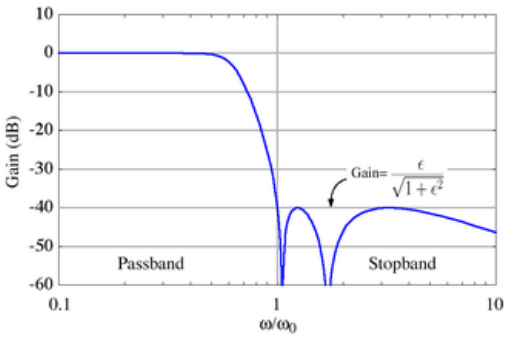

Figure 4.12 Frequency Response of Fifth-Order Type II Chebyshev Low-Pass Filter

Also known as inverse Chebyshev, this type is less common because it does not roll off as fast as type I, and requires more components. It has no ripple in the passband, but does have equiripple in the stopband. The gain is:

$$G_n(\omega, \omega_0) = \frac{1}{\sqrt{1 + \frac{1}{\varepsilon^2 T_n^2(\omega_0/\omega)}}}. \qquad (4.33)$$

In the stop band, the Chebyshev polynomial will oscillate between 0 and 1. The smallest frequency at which this maximum is attained will be the cutoff frequency ω_0. The parameter ε is thus related to the stopband attenuation γ in decibels by:

$$\varepsilon = \frac{1}{\sqrt{10^{0.1\gamma} - 1}}. \qquad (4.34)$$

The transfer function will be given by the poles in the left half plane of the gain function, and will have the same zeroes but these zeroes will be single rather than double zeroes.

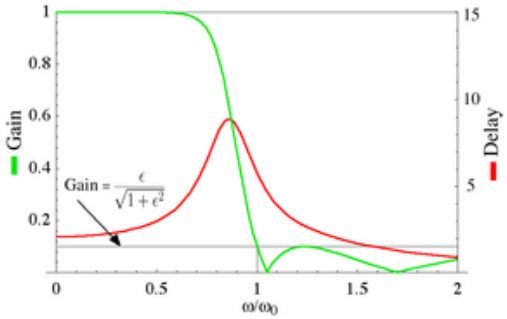

Figure 4.13 Gain and Group delay of Fifth-Order Type II Chebyshev Filter

The gain and the group delay for a fifth order type II Chebyshev filter with ε=0.1 are plotted in the graph. It can be seen that there are ripples in the gain in the stop band but not in the pass band.

4.7.5 Gaussian Filter

Gaussian filter is a filter whose impulse response is a Gaussian function. Gaussian filters are designed to give no overshoot to a step function input while minimizing the rise and fall time. This behavior is closely connected to the fact that the Gaussian filter has the minimum possible group delay. Mathematically, a Gaussian filter modifies the input signal by convolution with a Gaussian function; this transformation is also known as the Weierstrass transform.The one dimensional Gaussian filter has an impulse response given by:

$$g(x) = \sqrt{\frac{a}{\pi}} \cdot e^{-a \cdot x^2} \tag{4.35}$$

or with the standard deviation as parameter:

$$g(x) = \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma} \cdot e^{-\frac{x^2}{2\sigma^2}}. \tag{4.36}$$

In two dimensions, it is the product of two such Gaussians, one per direction:

$$g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4.37}$$

where x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution. When applied in two dimensions, this formula produces a surface whose contours are concentric circles with a Gaussian distribution from the center point. Values from this distribution are used to build a convolution matrix which is applied to the original image. Each pixel's new value is set to a weighted average of that pixel's neighborhood. The original pixel's value receives the heaviest weight (having the highest Gaussian value) and neighboring pixels receive smaller weights as their distance to the original pixel increases.

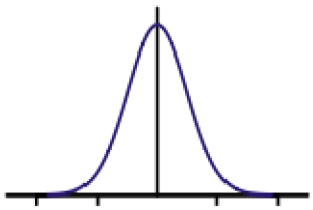

Figure 4.14 Shape of A Typical Gaussian Filter

DIGITAL IMPLEMENTATION

- Since the Gaussian function decays rapidly, it is often reasonable to truncate the filter window and implement the filter directly for narrow windows; in other cases this introduces significant errors, and one should instead use a different window function.
- The Gaussian kernel is continuous; most commonly the discrete analog is the sampled Gaussian kernel (sampling points from the continuous Gaussian), but the discrete Gaussian kernel is a better analog and has superior characteristics.
- Since the Fourier transform of the Gaussian function yields a Gaussian function, apply the Fast Fourier transform to the signal (preferably divided into overlapping windowed blocks), multiply with a Gaussian function and transform back. This is the standard procedure of applying an arbitrary finite impulse response filter, with the only difference that the Fourier transform of the filter window is explicitly known.
- Due to the central limit theorem approximate the Gaussian by several runs of a very simple filter like the moving average. The simple moving average corresponds to convolution with the constant B-spline, and e.g. four iterations of a moving average yields a cubic B-spline as filter window which approximates the Gaussian quite well. Interpret the standard deviation of a filter window as a measure of its size. For standard deviation σ and sample rate f obtain the frequency which can be considered the cut-off frequency. A simple moving average corresponds to a uniform probability distribution.

4.7.6 Optimum "L" Filter

The Optimum "L" filter (also known as a Legendre filter) was proposed by Athanasios Papoulis in 1958. It has the maximum roll off rate for a given filter order while

maintaining a monotonic frequency response. It provides a compromise between the Butterworth filter which is monotonic but has a slower roll off and the Chebyshev filter which has a faster roll off but has ripple in either the pass band or stop band. The filter design is based on Legendre polynomials which is the reason for its alternate name and the "L" in Optimum "L".

4.7.7 Linkwitz-Riley Filter

A Linkwitz-Riley (L-R) filter is an infinite impulse response filter used in Linkwitz-Riley audio crossovers, named after its inventors Siegfried Linkwitz and Russ Riley. It is also known as a Butterworth squared filter. An L-R crossover consists of a parallel combination of a low-pass and a high-pass L-R filter. The filters are usually designed by cascading two Butterworth filters, each of which has −3 dB gain at the cut-off frequency. The resulting Linkwitz-Riley filter has a −6 dB gain at the cutoff frequency. This means that summing the low-pass and high-pass outputs, the gain at the crossover frequency will be 0 dB, so the crossover behaves like an all-pass filter, having a flat amplitude response with a smoothly changing phase response. This is the biggest advantage of L-R crossovers compared to Butterworth crossovers, whose summed output has a +3 dB peak around the crossover frequency. Since cascading two nth order Butterworth filters will give a 2nth order Linkwitz-Riley filter. However, crossovers of higher order than 4th may have less useability due to their increasing peak in group delay around crossover frequency and complexity.

4.8 PROBLEM DEFINITION

The extracting of a signal buried in noise problem is stated as follows:

Noise is unwanted electrical or electromagnetic energy that degrades the quality of signals and data that is required, thereby, minimizing the effects of noise is very essential while also retaining the information or characteristic of the signal which is desired to be extracted from the signal.that is, preprocessing of signal is done to obtained the useful information from the noise and present it in a form more comprehensible than the raw data.

4.9 PROBLEM FORMULATION

The extraction of a signal buried in noise problem is formulated as an application of digital signal processing, that is, digital signals are processed to remove noise

disturbances and extract useful information. Various types of digital filters of different orders are used to obtain the desired signal within specific frequency band due to their superiority over analog filters, with better flexibility in processing for noise reductions. LabVIEW have many libraries with a large number of functions and advanced mathematic blocks for functions such as integration, filters, and other specialized capabilities usually associated with data capture from hardware sensors is immense such as data acquisition, signal generation, mathematics, statistics, signal conditioning, analysis, etc., along with numerous graphical interface elements which are used for the above problem defined.

CHAPTER - 5

RESULT AND DISCUSSION

LabVIEW has distinguished feature of effective preprocessing the signal to extract the useful information. To represent its typical application of data acquisition from noise, that is, unwanted random addition to a wanted signal. Using LabVIEW block diagram programming approach and extensive set of LabVIEW signal processing and measurement VI's simplify the development of analysis applications. In labVIEW, front panel can contain knobs, push buttons etc. which are the interactive input controls and graphs, LED's etc. which are indicators. The front panel objects have corresponding terminal on the block diagram. The various blocks are used in logic building in the block diagram which is the actual executable program. Firstly, the extraction of sine wave from noisy signal is shown to justify the importance of labVIEW in preprocessing of signal. For this uniform white noise is generated by using its VI that generates a uniformly distributed, pseudorandom pattern whose values are in the range [-a:a], where a is the absolute value of amplitude.

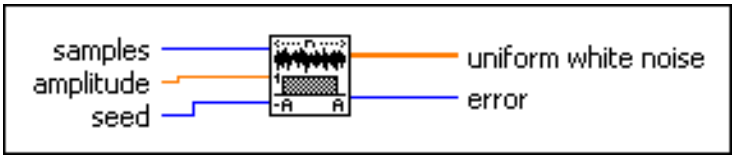

Figure 5.1 Uniform White Noise.vi

Where, samples is the number of the samples of the uniform white noise**.** samples must be greater than or equal to 0. The default is 128. Amplitude is the amplitude of uniform white noise. The default is 1.0. seed, when greater than 0, causes reseeding of the noise sample generator. The default is –1. If seed is less than or equal to 0, the noise generator is not reseeded and resumes producing noise samples as a continuation of the previous noise sequence**.** Uniform white noise contains the uniformly distributed, pseudorandom pattern. The largest uniform white noise that the VI can generate depends upon the amount of memory in your system and is theoretically limited to 2,147,483,647 ($2^{31} - 1$) elements. Error returns any error or warning from the VI. Then digital filter VI is used that generates a digital butterworth filter by calling the butterworth coefficients VI. After

calling the Butterworth Coefficients VI, the Butterworth Filter VI calls the IIR Cascade Filter VI to obtain a Butterworth Filtered X sequence. The values for high cutoff frequency and low cutoff frequency must observe the following relationship:

$$0 < f_l < f_2 < 0.5 f_s \qquad (5.1)$$

where f_l is low cutoff freq: f_l, f_2 is high cutoff freq: f_h, and f_s is sampling freq: f_s .

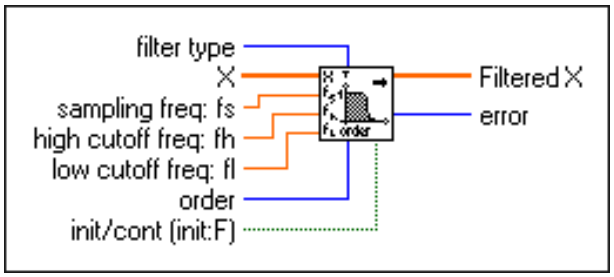

Figure 5.2 Butterworth Filter.vi

Here, filter order is proportional to the number of filter coefficients used in filtering operation. Increasing Filter Order improves filter's ability to attenuate frequency components in its stop band and reduces ripples in pass band. Then, sine pattern.vi is used to generate an array containing a sinusoidal pattern as shown in figure below:

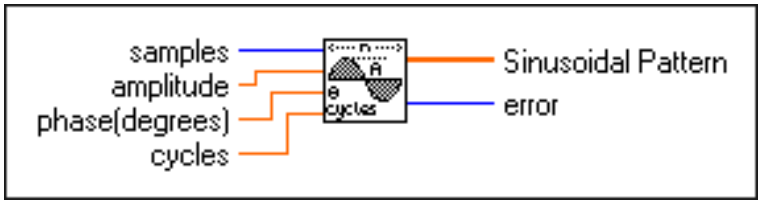

Figure 5.3 Sine Pattern.vi

The While Loop is used to repeat the subdiagram inside it until the conditional terminal, an input terminal, receives a particular Boolean value. The Boolean value depends on the continuation behavior of the While Loop. Bundle is used in this logic building to assemble the cluster from individual elements. Block wait(ms) also used to wait for the specified number of milliseconds and returns the value of the millisecond timer. Therfore, the execution is not completed until the specified time has elapsed. Waveform graphs are used to display the data that block diagram generates and basic blocks like add, reciprocal and boolean control are also used in logic building. The following figures shows the result obtained on front panel after the extraction of sine, square and triangular wave from noisy signal using labVIEW.

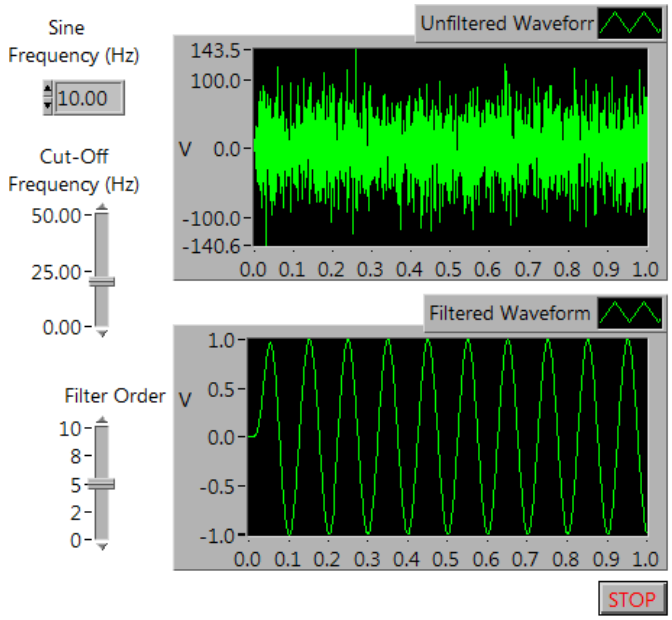

Figure 5.4 Sine Wave obtained after Filteration of Noisy Signal Using LabVIEW

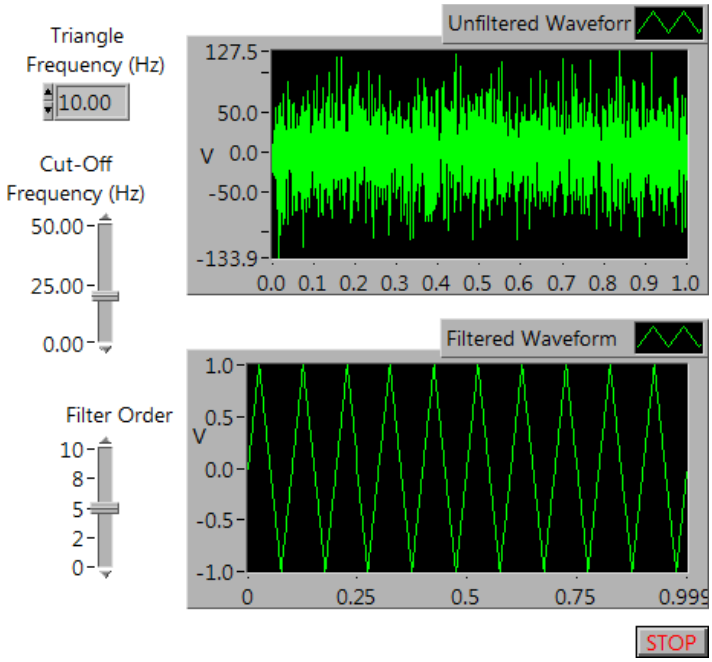

Figure 5.5 Triangular Wave obtained after Filteration of Noisy Signal Using LabVIEW

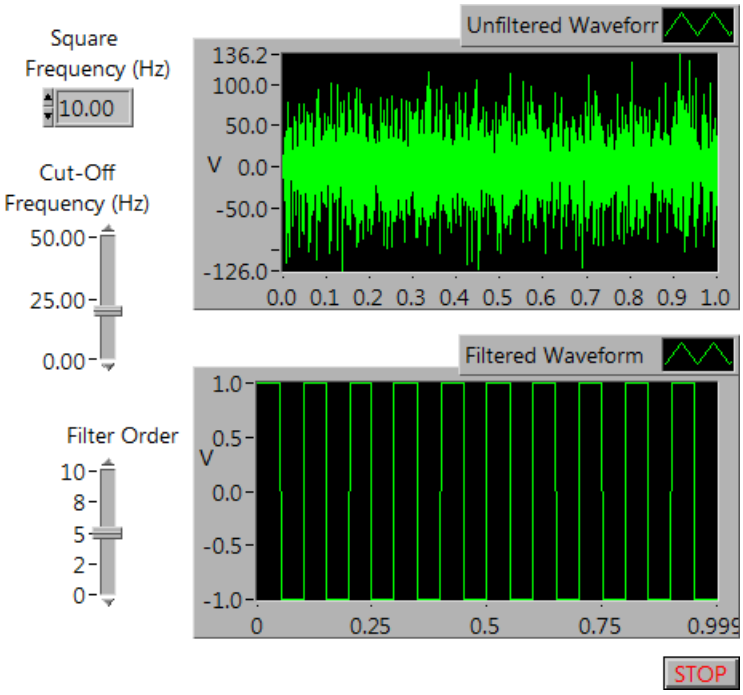

Figure 5.6 Square Wave obtained after Filteration of Noisy Signal Using LabVIEW

CHAPTER - 6

CONCLUSION AND FUTURE SCOPE

In this present work, using labVIEW for extraction of signal of different types from white noise with Butterworth Filter have obtained the desired results within specific frequency band. With data flow programming approach and many libraries of extensive set of signal processing VI's for various functions related to data acquisition, signal conditioning and signal analysis along with numerous graphical interface elements have improved the noise reduction and led to efficient extraction of desired signal. The benefit of LabVIEW over other development environments is the extensive support for accessing instrumentation hardware, thus, any signal type can be processed by using various digital filters of different orders for signal processing.

It is forseen that the speed of digital signal processing keep on increasing especially for high frequency signal better ways are to be devised for dealing with noise disturbances. More features are required to be added in processing software to develop efficient system for various application related to biomedical, instrumentation, information technology and communication. The improvement in the labVIEW signal processing VI's will lead to better signal generation and analysis along with efficient digital filtering to extract the useful information from specific frequency band, subsequently removing the unwanted noise.

REFERENCES

[1] J.B. Johnson; "Thermal Agitation of Electricity in Conductors", *Phys. Rev 32, 97, the Experiment,* 1928

[2] P.K. Dash, P.K. Nanda and S. Saha; "Artificial neural network and pattern recognition approach for narrowband signal extraction", *IEEE International Conference, vol. 1, pp. 282-292,* 1991

[3] Steven G. Mason, Gary E. Birch, and Mabo R. Ito; "Improved Single-Trial Signal Extraction of Low SNR Events", *IEEE Transactions on signal processing, vol. 42, no. 2, February,* 1994

[4] Giacinto Gelli and Luigi Paura; "Blind Signal Extraction Based on Higher-Order Cyclostationarity Properties", *IEEE International Conference, vol. 5, pp. 3473-3476,* 1997

[5] N. Hemel and J. Leski; "HRV Signal Extraction Using Multirate Approach", *Proceedings of The first Joint BMES/EMBS Conference Serving Humanity, Advancing Technology, Atlanta, USA,* 1999

[6] Jiann-An Tsai, J. Hicks and Brian D. Woerner; "Joint MMSE Beam forming with SIC for an Overloaded Array System", *IEEE International Conference, vol. 2, pp. 1261-1265,* 2001

[7] Xiaohuan Li and Shan Ouyang; "One Reduced-Rank Blind Fresh Filter for Spectral Overlapping Interference Signal Extraction and DSP Implementation", *IEEE International Workshop on Intelligent Systems and Applications, pp. 1-4,* 2003

[8] J.M. Graybeal and M.T. Petterson ; "Adaptive Filtering and Alternative Calculations Revolutionizes Pulse Oximetry Sensitivity and Specificity During Motion and Low Perfusion", *Proceeding of American Control Conference, vol. 6, pp. 3677-3684,* 2004

[9] Sergio A. Cruces-Alvarez and Shun-ichi Amari; "From Blind Signal Extraction to Blind Instantaneous Signal Separation: Criteria, Algorithms, and Stability", *IEEE Transactions on Neural Networks, vol. 15, no. 4,* 2004

[10] Hagit Messer; “Analysis of the Input Noise Contribution in the Noise Temperature Measurements”, *IEEE Transactions on Industrial Electronics, vol. 47, no. 1, pp. 193-202,* 2005

[11] Muhammad A. Hasan and Muhammad I. Ibrahimy; “VHDL Modeling of FECG Extraction from the Composite Abdominal ECG using Artificial Intelligence”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B, vol. 30, no. 4, pp. 510-516,* 2006

[12] Zhiqiang Gao and P. C. Ching; "White noise in MOS transistors and resistors", *IEEE Circuits Devices Magazine, pp. 23–29,* 2006

[13] Keiji Watanabe and Chih-Kuo Liang; "The Determination of Noise Temperatures of Large Paraboloidal Antennas”, *IEEE Transactions on Antennas and Propagation, vol. 10,* 2006

[14] Kianoush Nazarpour, Martin Stridh and W. Bequette; “Comparison of Atrial Signal Extraction Algorithms in 12-Lead ECGs With Atrial Fibrillation”, *IEEE Transactions on Biomedical Engineering, vol. 53, no. 2,* 2007

[15] U Fai Chan and Wai Wong Chan; “Flexible Implementation of Front-end Bioelectric Signal Amplifier using FPAA for Telemedicine System”, *Proceedings of the SICE annual conference Fukuri, vol. 1, pp. 187-190,* 2007

[16] Philip Langley and Frenzer Patterson; “A Ultrasonic Signal Detection Via Improved Sparse Representations”, *Proceedings of 2007 International Conference on Machine Learning and Cybernetics, vol.2, pp. 983 - 987,* 2007

[17] Guihua Han, Lihua Chen and QI Ai-ling; “Blind source separation of electrocardiographic signals using system stability criteria”, *Proceedings of the 29th Annual International Conference of the IEEE EMBS, Lyon, pp. 1228-1232,* 2007

[18] Jorge E. Monzon and D. V. Perov; “Extraction of Signals from White and Colored Noise Using the Discrete Wavelet Transform”, *Acoustical Physics, vol. 51, no. 4, pp. 443–448,* 2007

[19] X.W. Gao, Stephane Rossignol and Toshihisa Tanaka; “Extraction of Steady State Visually Evoked Potential Signal and Estimation of Distribution Map from EEG Data”, *Proceedings of the 29th Annual International Conference of the IEEE EMBS, Lyon, pp. 228-232,* 2007

[20] Rahman Jamal, Mike Cerna and John Hanksh; “Designing Filters Using the Digital Filter Design Tools in the Signal Processing Toolsets”, *National Instruments, September* 2007

[21] E. Vassalos and E. Zigouris; “LabVIEW to CCS Link for Automating Digital Signal & Image Processing Applications”, *Proceedings of the ICSES'04, International Conference on Signals and Electronics Systems, pp.569-572, Poland,* 2008.

[22] Jani Even and Hiroshi Saruwatari; “Blind Signal Extraction Via Direct Mutual Information Minimization”, *Proceedings of IEEE International Conference on Automation and Logistics, pp. 881-885,* 2008

[23] Istvan A.Szabo and Lajos Harasztosi; "Ways to use LabVIEW to aid DSP education", *Proceedings of the EDERS-2008, 2nd European DSP Education & Research Symposium, Munich, Germany,* 2008.

[24] Nasser Kehtarnavaz and Namjin Kim; "Digital Signal Processing System-Level Design Using LabVIEW", *Elsevier,* 2008

[25] M. Babaie-Zadeh, C. Jutten, and K. Nayebi; “Differential of the mutual information,” *IEEE Signal Processing Letters, vol. 11, no. 1, pp. 48–51,* 2008

[26] R. R. Gharieb and A. Cichocki; “ Noise reduction in brain evoked potentials based on third-order correlations”, *IEEE transactions on Biomedical Engineering, vol. 48, no. 5, pp. 501–512,* 2009