2012-08-27

IMU design for high vibration environments with special consideration for vibration rectification

Zaiss, Curtis

Department of Mechanical and Manufacturing Engineering

IMU Design for High Vibration Environments with Special Consideration for Vibration Rectification

by

Curtis Zaiss

UNIVERSITY OF CALGARY

IMU Design for High Vibration Environments with Special Consideration for Vibration Rectification

by

Curtis Zaiss

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE IN ENGINEERING

DEPARTMENT OF MECHANICAL AND MANUFACTURING ENGINEERING

CALGARY, ALBERTA

AUG 2012

ABSTRACT

Inertial measurement units (IMU)s have been in use for decades. When first developed the uses were usually limited to space travel and military applications. With the development of new technologies and new sensors IMUs are being used in many more applications like street mapping, airborne surveying, down-hole logging and directional drilling, unmanned vehicles, and even in smart phones. The variety of applications has lent itself to a wide variety of sensors being developed with costs ranging from a few dollars to few thousand dollars. While the development of these sensors and IMUs has led to more widespread uses and a general improvement in the performance to cost ratio, there has remained a gap in understanding of the effects of vibration on the accuracy of the system. The result of this gap is poor performance under vibration. Both the designer and user must be made aware of the vibration related errors to avoid them.

This thesis explores the effects of nonlinearity, system noise, computational error, filtering, and system dynamics with the goal of providing an understanding of the errors and their interactions when subjected to vibration or high dynamics. By providing methods of analyzing and in many cases compensating for errors, it is possible to design a more robust system.

ACKNOWLEDGEMENTS

I would like to express my appreciation to my supervisor, Dr Swavik Spiewak for his encouragement and enthusiasm. I would also like to thank Kaj and Denise Hedin from Special Projects Inc. whose support allowed me to do this.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOTATIONS, SYMBOLS AND ACRONYMS

$1\,Gal = 0.01 m/s^2 = 0.001g$

$1\,mGal = 10^{-5}\,m/s^2 = 1\mu g$

Acronyms	
VRE	Vibration Rectification Error
THD	Total Harmonic Distortion
IMU	Inertial Measurement Unit
MEMS	Micro Electro Mechanical Systems
GPS	Global Positioning System
Q-Flex	Type/Brand of accelerometer characterized by Quartz Flexures, pendulous proof mass support, force rebalanced servo control
FPGA	Field Programmable Gate Array
ADC	Analog to Digital Converter
SINS	Strapdown Inertial Navigation System
UAV	Unmanned Aerial Vehicle
ENOB	Effective Number of Bits

Symbol	**Meaning**	**Section**
a_{out}	Accelerometer output	2.4
b	Accelerometer bias	2.4
$\boldsymbol{S}$	Scale Factor	2.4
a_{sf}	Specific Force Acceleration	2.4
G_{aln}	Alignment matrix	2.4
$S_0, S_1, S_2 \ldots S_n$	Terms of nonlinearity	2.4
T	temperature	2.4
t	time	2.4

Symbol	Meaning	Section
θ	Attitude of body relative to inertial frame	3.1.1
ω_{yb}	Angular Rate of Body relative to inertial frame	3.1.1
V	Velocity	3.1.1
f_{xi}	Specific Force in x axis of inertial frame	3.1.1
f_{zi}	Specific Force in z axis of inertial frame	3.1.1
f_{xb}	Specific Force in x axis of body frame	3.1.1
f_{zb}	Specific Force in z axis of body frame	3.1.1
$g_{xi}\ g_{zi}$	Gravity in x and z axis of inertial frame	3.1.1
$V_{xi}\ V_{zi}$	Velocity in x and z axis of inertial frame	3.1.1
$X_i\ Z_i$	Position in the x and z axis of the inertial frame	3.1.1

Symbol	Meaning	Section
$\boldsymbol{\omega}$	Angular Rate Vector	3.1.2
$\boldsymbol{f}^b$	Specific Force, Body Frame	3.1.2
$\boldsymbol{f}^n$	Specific Force, Navigation Frame	3.1.2
$\boldsymbol{g}^n$	Gravitiy, Navigation Frame	3.1.2
$\boldsymbol{C}_b^n$	Rotation Matrix, Converts Body Frame to Nav. Frame	3.1.2
$\boldsymbol{\omega}_{ie}^n$	Earth rotation rate, Nav. Frame	3.1.2
$\boldsymbol{\omega}_{en}^n$	Transport Rate, Rate of Change of Nav. Frame WRT Earth Frame as seen in Nav. Frame	3.1.2
$\boldsymbol{\omega}_{ib}^b$	Angular Rate measurements in Body Frame. From Gyro	3.1.2
$\boldsymbol{\omega}_{nb}^b$	Angular rate of body relative to navigation frame, in body frame	3.1.2
$\boldsymbol{v}^n$	Velocity in Nav. Frame	3.1.2
$\boldsymbol{C}_n^b$	Rotation Matrix, Converts Nav. Frame to body Frame	3.1.2
$\boldsymbol{\Omega}_{\mathrm{nb}}^{\mathrm{b}}$	Skew symmetric of $\boldsymbol{\omega}_{nb}^b$	3.1.2
ψ, θ, ϕ	Yaw, Pitch and Roll	3.1.2

Symbol	Meaning	Section
$\mu g/g^2$	Units of Measure for vibration rectification	3.2.2
$y(x)$	Output of a nonlinear system with input x	3.2.2
$y(t)$	Output of a nonlinear system as function of time	3.2.2
x or $x(t)$	Input to nonlinear system	3.2.2
$a_0, .. a_n$	Coefficients of nonlinearity	3.2.2
r	Magnitude of sinusoidal excitation	3.2.2
w	Frequency of sinusoidal excitation	3.2.2
b_0	Zeroth order coefficient of trigonometric fourier series	3.2.2
b_n	nth order coefficient of trigonometric fourier series	3.2.2
T	Period of oscillation	3.2.2

Symbol	Meaning	Section
ΔT	Temperature rise due to self-heating	3.2.5
ρ	Thermal resistance to heat sink	3.2.5
$P(a)$	Power dissipation as function of acceleration	3.2.5
τ	Thermal time constant	3.2.5
t	Time since onset of acceleration	3.2.5

Symbol	Meaning	Section
SNR_{ideal}	Signal to Noise ratio for ideal ADC with Jitter	3.2.6.1
f	Frequency of Sampling	3.2.6.1
t_j	Period of Clock Jitter	3.2.6.1
SNR_{bit}	Signal to Noise Ratio for ADC with N bits	3.2.6.1
N	Number of Bits	3.2.6.1
SNR_{loss}	Signal to Noise Ratio Loss due to amplifier noise	3.2.6.2
V_n	Noise of ADC $\mu V\ rms$	3.2.6.2
FSR	Full Scale Range Volts	3.2.6.2
e_n	Input noise spectral density $nV/\sqrt{Hz}$	3.2.6.2
G	Amplifier Gain	3.2.6.2
I_q	Quiescent Current	3.2.6.3
E_{th}	Thermal Noise Volts	3.2.6.3
k	Boltzmann's constant	3.2.6.3
T	Temperature	3.2.6.3
R	Resistance	3.2.6.3
B	Noise Bandwidth	3.2.6.3
E_n	Voltage Noise	3.2.6.4
f_{nc}	Corner Frequency for 1/f noise	3.2.6.4
$f_H\ \ f_L$	Highest and Lowest frequencies in bandwidth	3.2.6.4
E	Normalized Error Due to Delay	3.2.6.4
f	Frequency of oscillation, excitation, vibration	3.2.6.4
t_d	Period of Delay	3.2.7

Symbol	Meaning	Section
$\boldsymbol{C}$	Direction Cosine Matrix	3.4.1
$\boldsymbol{\Omega}$	Skew Symmetric Angular Rates	3.4.1
$\boldsymbol{\sigma}\times$	Skew Symmetric of Rotation Vector	3.4.1
$\boldsymbol{\sigma}$	Rotation Vector	3.4.1
σ	Magnitude of rotation vector	3.4.1
$\boldsymbol{A}_k$	Direction Cosine Matrix which transforms C_k to C_{k+1}	3.4.1
a_1 and a_2	Coefficients of Bortz Equation	3.4.1
D_{dc}	DC angular drift error	3.4.1

Symbol	Meaning	Section
z	Measurement vector	4.2
$\boldsymbol{H}$	Observation Matrix, also known as design matrix, model matrix, measurement coupling matrix	4.2
x	State vector, or vector of unknowns	4.2
ε	Measurement error	4.2
$\hat{x}$	Estimate of state vector or unknowns	4.2
$\boldsymbol{\mathcal{G}}$	Gramian Matrix	4.2
$\boldsymbol{R}$	Observation weighting matrix, or measurement covariance matrix	4.2
$\boldsymbol{K}$	Kalman Gain matrix	4.2
$\boldsymbol{P}_1$	Process covariance matrix	4.2
x	State vector	4.2
$\boldsymbol{F}$	Dynamics matrix	4.2
$\boldsymbol{B}$	Input coupling matrix	4.2
u	Input vector	4.2
$\boldsymbol{G}$	Noise coupling matrix	4.2
w	Noise vector	4.2

$\boldsymbol{\Phi}_{k,k+1}$	State Transition matrix which propagates the state transition matrix from time k to k+1	4.2
$\boldsymbol{P}_k^-$	Priori error covariance	4.2
$\boldsymbol{P}_k^+$	Posteriori error covariance	4.2
$\boldsymbol{Q}$	Discrete system noise matrix	4.2
v_{ref}	Velocity reference	4.3
a_{ref}	Acceleration Reference	4.3
$c_0 \dots c_n$	Coefficients of nonlinearity that linearizes the signal	4.3
$b_1 \dots b_n$	Coefficients of polynomial function describing the bias drift, used in least squares	4.3
v	velocity	4.3
b_{rw}	Bias random walk	4.3
$\boldsymbol{Q}_k$	Discrete process noise matrix	4.3

CHAPTER 1: INTRODUCTION

1.1 Background

The recent shift to low cost sensors has led to an exponential growth in the range of applications for inertial measurement. Some examples are airborne surveying, airborne gravimetry, down-hole directional drilling and logging, unmanned aerial vehicles, vehicle navigation, personal navigation, attitude referencing, camera stabilization, etc. Many if not all of these examples are situations where vibration is present at some level. This vibration when combined with various sensor and system errors causes vibration rectification and harmonic distortion.

According to standard definitions, vibration rectification is an apparent shift in acceleration due to sinusoidal excitation [1]. While this is an accurate definition, a broader definition is needed to account for all error sources. Vibration rectification is any DC error caused by a systems response to sinusoidal input. Vibration rectification errors are often dependent on the vibration patterns. As these patterns change, so too do the errors. This change often takes place before the errors can be quantified [2]. As a result the only reliable method of resolving the error is to compensate for them directly rather than quantify them over time.

Vibration can in many ways be thought of as high rate dynamics. In this way the requirements for measurement under vibration are similar to those already described in literature. Many sources stress the requirement for high sampling and processing speeds when under high dynamic rates. The compensation of nonlinearity, which is a key source of vibration rectification error, is among the primary means of reducing the error however it is not commonly used. Many systems filter the sensor outputs to eliminate vibration

altogether which does not allow for the detection or removal of VRE. Few inertial systems are even capable of measuring the vibration and performing the computations at the required rate. This has led to a tendency to ignore vibration related errors relying instead on aiding devices to correct for error buildup.

1.2 Problem Description

This research seeks to create an IMU for high vibration environments. This has been driven mainly by requirements for airborne surveying in locations where GPS coverage is not ideal and where aircraft attitude during turns causes outages. The clear answer is inertial measurement integrated with GPS. While this has improved positioning accuracy, problems still exists with the inertial measurement under certain conditions. The dynamic and vibratory environment caused by the aircraft's engines, turbulence, and high rate maneuvers cause the inertial measurement to be unreliable due to vibration related errors. Inability to measure the dynamics and vibration fully results in an inability to determine the root cause of the errors.

The objective of this research is to develop an inertial measurement unit and data acquisition system capable of accurate measurement under high dynamics and vibration. To achieve this it is necessary to investigate vibration rectification and other vibration related errors in order to better understand the requirements for system design, calibration and compensation. Furthermore, a system is required for the analysis and evaluation of compensation techniques and inertial integration. To allow for the evaluation of different algorithms or hardware, this system should be highly adaptable.

1.3 Thesis Outline

The thesis is divided into 6 chapters. Chapter 2 reviews related technology, sensor errors and related research. Chapter 3 provides an in depth coverage of data acquisition and sensor errors, focusing of vibration related errors. Examples are provided and recommendations are given for avoidance or compensation of many of these errors. Chapter 4 introduces an alternate calibration method for nonlinearity, as well as formulations for the estimation of parameter using least squares and Kalman filtering. Chapter 5 presents the design for an IMU data acquisition system. Finally, Chapter 6 presents conclusions and recommendations for future work.

The main contributions in this thesis are:

- Improved definition of vibration rectification error
- Development of test and calibration methods to easily identify and compensate for nonlinearity and vibration rectification.
- Discussion of numerous system design issues which affect the performance of inertial measurement units. Focus is given to errors which are caused by vibration.
- Development of an IMU system designed specifically for operation in high vibration or high dynamic environments.

CHAPTER 2: REVIEW OF RELATED TECHNOLOGY AND RESEARCH

2.1 Inertial Navigation Systems

Although not a navigation system, the gyrocompass was among the first practical implementations of an inertial sensor used for navigation aiding. World War II saw the introduction of inertially guided rockets [4]. Development of the 50s and 60s were focused on the stabilized platform navigation system intended for missiles, aircraft and space vehicles. In this system a set of gimbals are forced to maintain a fixed attitude with respect to the navigation frame. One of the benefits of this method is that the gyros need not measure the full rate. In fact the gimbal system is driven to practically eliminate the rate measurement all together. As a result non linearity and scale factor issues in the gyros are irrelevant. The other advantage is that the accelerometers are maintained in a particular direction where their behavior can be accurately modeled. The net result is still among the most accurate navigation systems to date.

While the stabilized platform is highly accurate, its downfalls are size, complexity, cost, and ruggedness. Almost since the original development of inertial navigation, the idea of a strapdown inertial system has been seen as the solution to these problems. Strapdown systems have added complications, one of which being the gyros are now required to measure the full angular rate accurately and the accelerometers are not maintained level. The leveling of the accelerometers is done analytically. Originally many of the computations were done using analog computers. Strapdown inertial systems have in recent decades become more feasible in part due to the improvements in sensor technology but mainly due to the improvements in computers.

The relatively new developments of MEMS inertial systems has lowered the cost and size of the systems opening the door to more applications. This latest shift has in some cases led to complacency with regards to the errors which have been well known for decades [2]. This disregard for common errors results in performance which is very poor in some conditions.

2.2 Ideal Strapdown Inertial System

Newer Low Cost IMUs are being created for numerous applications from vehicle navigation to UAVs. The priority and focus of the design of these systems is on cost and in many cases power and size reduction. These priorities have, according to Farrel [2], led to tendency to overlook many of the system errors which have been known for decades. Among these are the g-sensitive and vibration sensitive errors. It is Farrel's opinion that these errors are even more prevalent in low cost sensors and should be given greater emphasis. He lays out his ideal system design goals, some of which are:

- Accuracy commensurate with the best constituent subsystem
- Higher Sample Rates
- Versatility
- Maintainability
- Precise Time Tagging
- Adaptability

Of key importance to him is the availability of raw data throughout the system which enables many more of these traits to be realized. The author also hints to the possibility in the future of massively parallel processing in inertial instruments as a means of improving performance.

From recommendations and analysis given by [4], [5], [2], [3] the biggest issue with vibration and high dynamics seems to be sampling and processing rates. A quote from [4] seems to exemplify the concept. “The bandwidths of the sensors and the speed of the computations must be high enough to sense and record the actual motion, otherwise significant errors can arise, even if ‘perfect’ sensors were available.”

2.3 Accelerometer Technology

Although many acceleration measurement devices exist, their scale usually allows them to be defined as either mechanical or solid-state [6] [4]. Mechanical sensors have been developed since the 1950s where solid state sensors are relatively new. Recent development has led to MEMs devices with similar specifications to that of mechanical sensors as is shown in Figure 2.1.

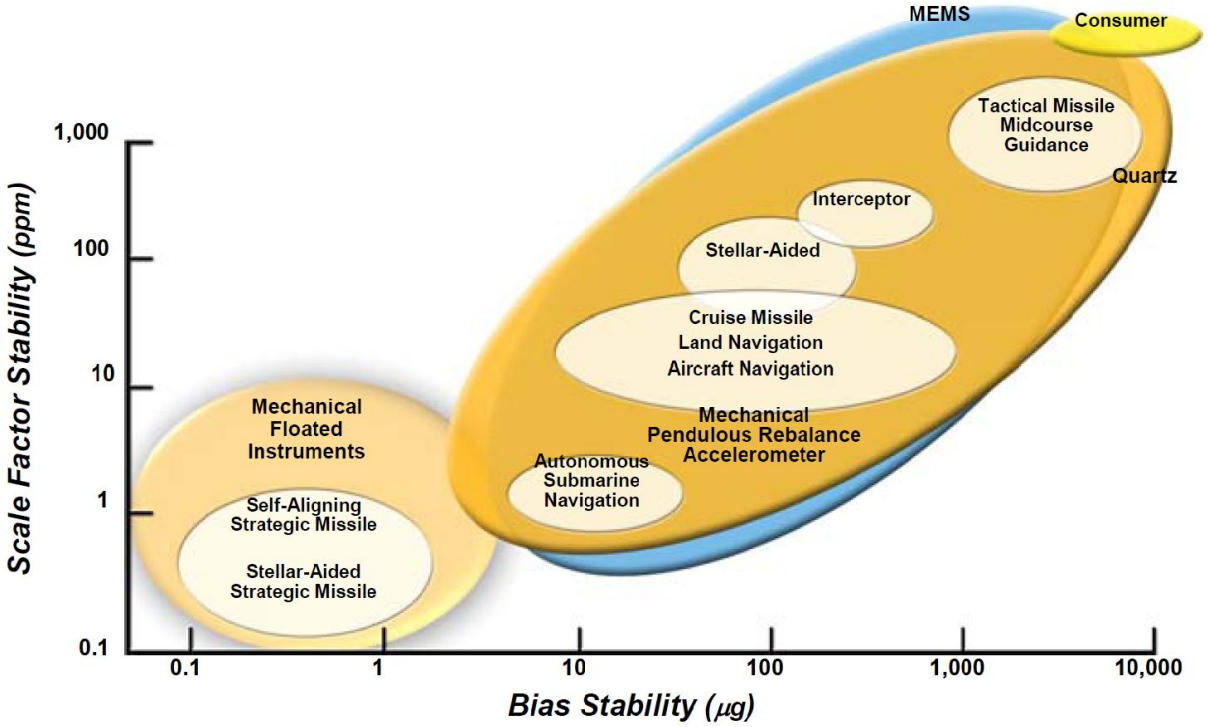

Figure 2.1 Bias and Scale Factor Stability [7]

2.3.1 Proof Mass Accelerometer

Proof mass accelerometers are among the simplest devices and most common. Essentially they measure the deflection of a proof mass inside the sensor body. In some cases a rebalance force is applied to prevent the displacement with respect to the case. These force rebalanced sensors usually output a signal proportional to that force rather than the displacement.

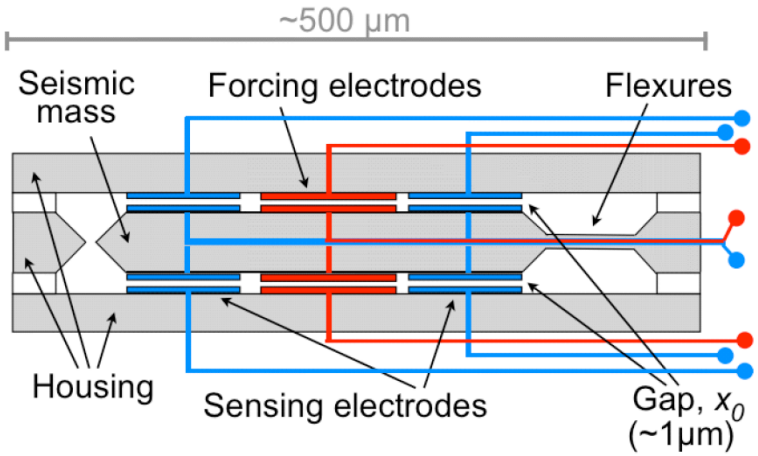

Figure 2.2 Capacitive Servo Accelerometer [8]

Figure 2.2 shows an example of a MEMS accelerometer using a pendulous proof mass and capacitive sensing and forcing in a servo loop. A similar sensor can be made open loop by not including the rebalance and servo mechanism.

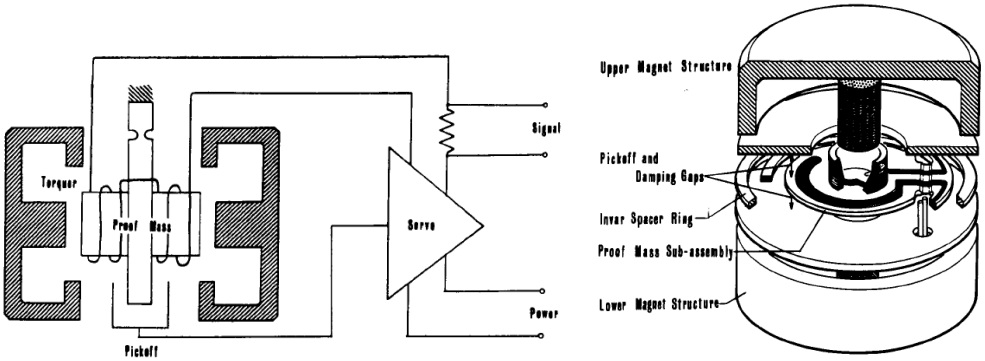

Figure 2.3 Q-Flex Servo Mechanism [9]

While the MEMS sensor shown above uses capacitive sensing and driving which is possible at the small scale, the Q-Flex sensor shown in Figure 2.3 uses an electromagnetic torquer coil to rebalance the proof mass. Apart from this difference the structure and operation of the two sensors is very similar. The proof mass, flexure and flexure support of the Q-Flex are formed from a single fused quartz blank [9]. Although the Q-Flex sensor was originally designed over 40 years ago, variations of this sensor are still sold with few differences in the overall design. They are in many ways the standard by which many navigation grade sensors are compared.

2.3.2 Pulse Integrating Gyro Accelerometer (PIGA)

In a PIGA device an integrated acceleration measurement is derived using a single degree of freedom gyro. The measurement is obtained by applying the acceleration along the input axis of the gyro. This force produces a precession in the gyro. The integral of the acceleration is proportional to the rotation angle of the vertical cylinder with respect to the outer case.

2.3.3 Vibrating String

A vibrating string accelerometer is essentially a proof mass accelerometer which measures the displacement differently. These accelerometers use two thin metal tapes with a sliding mass between them. As acceleration is applied in the sensitive axis, the frequency of the two strings change due to the differences in tension. The acceleration output is proportional to the difference in these frequencies.

2.3.4 Fiber Optic

Fiber optic accelerometers are another form of proof mass sensor. In this case the force required to accelerate the mass causes micro bends in the optical fiber. These bends modulate the intensity of the light and provide the output signal.

2.4 Accelerometer Errors

Regardless of the sensor type and cost, every accelerometer has errors. Many have the same errors, differing only in the character and magnitude. A nearly complete listing of the accelerometer errors and calibration methods can be found in [1]. Below is a short list of the common errors described with their test.

Table 2.1 IEEE Accelerometer Errors and Calibration Procedures

Error	Test/Calibration Procedure
Bias Stability	• Long term stationary measurement • Tumbling
Bias Temperature Sensitivity	• Centrifuge (in Oven) • Tumble tests (in Oven)
Scale factor Stability Temperature Sensitivity	• Centrifuge (in Oven) • Tumble tests (in Oven)
Vibration Rectification	• Vibration test
Nonlinearity	• Centrifuge test
Intrinsic Noise	• Long term stationary measurement
Axis Misalignment	• Tumble tests

In order to convert the sensor output of an accelerometer to specific force a model is used which accounts for these imperfections. The complexity and completeness of the sensor error models is usually determined by the user based on application. The models are usually very similar between Gyros and Accelerometers. A basic representative model is shown below.

$$\boldsymbol{a_{out} = b + S\, a_{sf} + noise} \qquad \textbf{(2.1)}$$

Where $\boldsymbol{a_{sf}}$ represents the specific force acceleration, **b** is the bias, **S** is the scale factor, and $\boldsymbol{a_{out}}$ is the accelerometer output. The above sensor model is the most basic version and is used only when the precision of the measurement is not highly important. For higher precision applications terms are added to compensate for nonlinearity, cross axis g sensitivity, bias sensitivity to temperature, scale factor sensitivity to temperature, axis

misalignment, and non-orthogonality. A possible model is represented by the following equations

$$a_{out} = G_{aln}\left(S_0 + S_1 a_{sf} + S_2 a_{sf}^2 \ldots S_n a_{sf}^n\right) \quad (2.2)$$

$$S_0, S_1, S_2 \ldots S_n = f(T, a_{sf}, t) \quad (2.3)$$

where G_{aln} is the alignment matrix, $S_0 \ldots S_n$ are coefficients of the nonlinearity of the acceleration. Equation (2.3) also expands on Equation (2.2) to say that every coefficient is also a function of temperature, specific force, and time.

2.5 Mechanization Algorithms

The process of mechanization is used to compute the position, velocity and attitude from the sensor measurements. The development of mechanization algorithms has followed closely the development of inertial systems. The optimization of the algorithms to minimize computing requirements was driven mainly by the limitations of the technology of the 50s and 60s. Some systems resorted to performing parts of the calculations with analog computers. One example of this was developed by Bortz [10].

Many sources of information exist for strapdown inertial mechanization. A handful are referenced here [6] [2] [4] [5] [11]. To allow for high frequency dynamics, high speed computation is required. Performing all computations at this rate presents a significant burden on the computer. A common technique in fully digital implementations uses multi-rate computation. These algorithms carefully weigh the accuracy of each computation based on the rate of change of the signals. One such method is shown below.

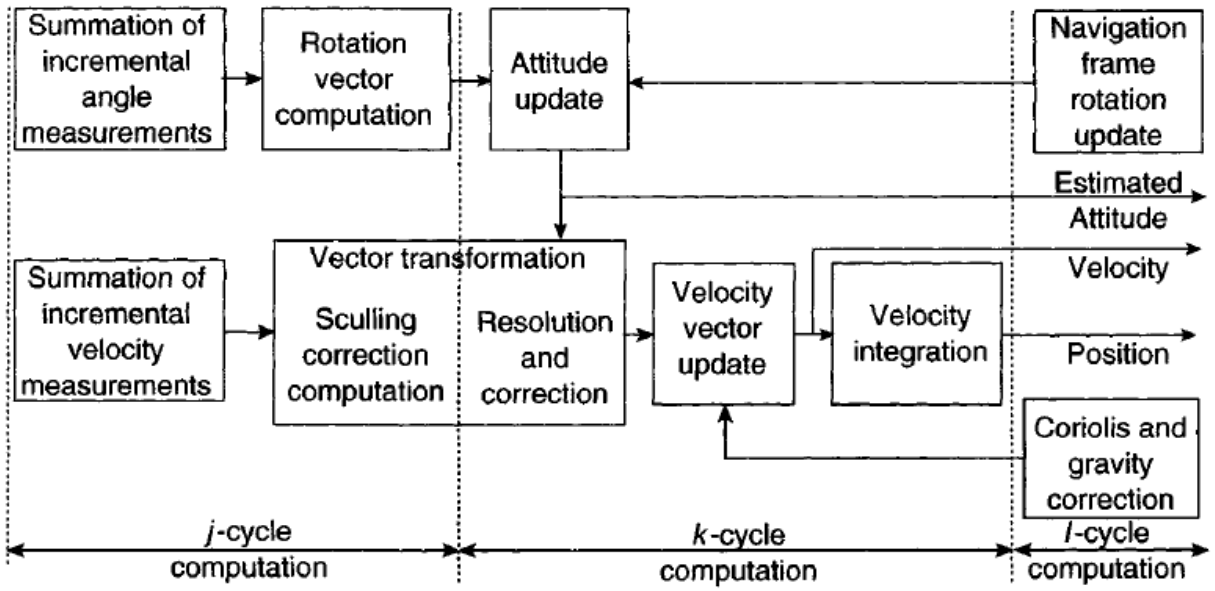

Figure 2.4 Multi-Rate Strapdown Mechanization [4]

Algorithms such as this one are still used in current systems as they reduce the computational requirements while maintaining accuracy.

2.6 Airborne Gravity Measurement Using Strapdown INS

The concept of gravity measurement using airborne gravity meters has only recently become viable with the aid of high accuracy GPS. The local gravity measurement is taken as the difference in the acceleration determined by GPS and by the accelerometer. The minimum precision required for useful measurements is on the order of 10 mGal or lower which equates to 10 μg.

The PhD theses, from [12] and [13], describe the development of an airborne gravity meter using a strapdown inertial navigation system (SINS) which is compared to a Lacoste Stable platform gravity meter. The size and cost advantages are immense. Where the Lacoste

meter is almost a cubic meter in volume, the SINS system is roughly a cubic foot. The weight and power reduction is significant as well.

The SINS system was built using various GPS receivers as well as a Honeywell Laseref III IMU which uses QA2000 accelerometers sampled at 50Hz. Both theses discuss the bias stability and random walk of the QA2000 but do not make any reference to nonlinearity or vibration rectification.

The thesis from Bruton [12] discusses the use of INS/GPS integration for the measurement of the local gravity field using airborne techniques. This thesis discusses problems with uncompensated bias in the accelerometer. His results and the results of Glennie [13] indicate uncompensated bias errors of the 10 to 20 mGal magnitude which seems to worsen in flight. At the time of publication neither had determined the cause of the errors.

From the analysis of the errors as described by the authors, it is possible to postulate that these errors may have been caused by a combination of vibration rectification, nonlinearity, time synchronization and thermal drift.

CHAPTER 3: ENHANCEMENT OF INERTIAL MEASUREMENT UNDER VIBRATION

3.1 Introduction

In order to fulfill the requirement of designing an IMU which has better accuracy under vibration, a complete analysis is required to determine how inertial sensors are used in the measurement, how the errors accumulate, and what types of errors are most affected by vibration. In this chapter this investigation begins with a review of the mechanization of the inertial measurement. This process converts raw measurements of acceleration and angular rates into position, velocity and attitude. The next stage is an investigation into the sensor related errors. At this stage emphasis is given to vibration related errors such as vibration rectification error from nonlinearity, self-heating, and integration. As the ADC and signal conditioning circuits are an integral part of the inertial data acquisition, an analysis of the noise, resolution, and stability is conducted to determine its contribution to system errors.

The next section demonstrates vibration rectification error from nonlinearity, and numerical integration. It also demonstrates the compensation method and provides examples for optimal and sub-optimal results. The final part of this investigation looks again at the mechanization to determine the impact of computational approximations when subjected to high rate movements.

3.1.1 Two Dimensional Strapdown Inertial Navigation

As an introduction, a simplified example is considered to demonstrate the principles. If a strapdown INS is restricted to navigate in a flat vertical plane, it requires two accelerometers and one gyro [4]. Figure 3.1 demonstrates the arrangement of sensors as well as the schematic diagram of the computation for position, velocity and attitude.

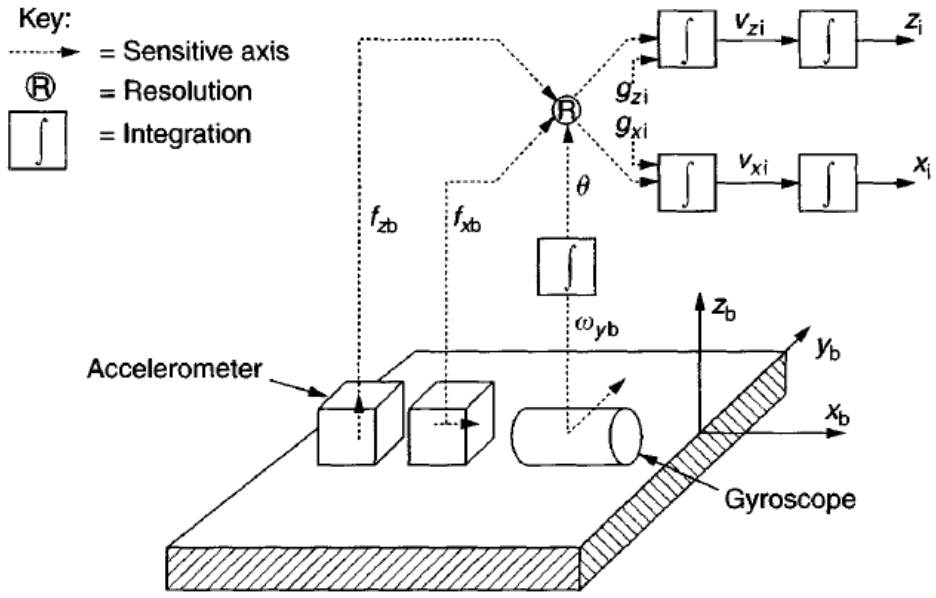

Figure 3.1 Two Dimensional Strapdown Navigation [4]

The following set of equations is sufficient to fully determine the position $\{X_i, Z_i\}$ and attitude θ of the system above in the inertial reference frame.

$$\dot{\theta} = \omega_{yb} \quad \textbf{(3.1)}$$

$$f_{xi} = f_{xb}\cos(\theta) + f_{zb}\sin(\theta) \quad \textbf{(3.2)}$$

$$f_{zi} = -f_{xb}\sin(\theta) + f_{zb}\cos(\theta) \quad \textbf{(3.3)}$$

$$\dot{V}_{xi} = f_{xi} + g_{xi} \quad \textbf{(3.4)}$$

$$\dot{V}_{zi} = f_{zi} + g_{zi} \quad \textbf{(3.5)}$$

$$\dot{X}_i = V_{xi} \quad \textbf{(3.6)}$$

$$\dot{Z}_i = V_{zi} \quad \textbf{(3.7)}$$

Equation 3.1 constitutes the attitude update equation for a single axis. Equation 3.2 and 3.3 represent the conversion of the body frame accelerations to inertial frame. The remainder of the set allow for the determination of velocity and position in the inertial frame.

3.1.2 Three Dimensional Strapdown Navigation

When navigating in the vicinity of Earth, a number of reference frames are used to compute position. The following reference frames are used in the equations to follow.

- i-Frame: Inertial Non Rotating
- e-Frame: Earth Centered Earth fixed
- n-Frame: Navigation Frame
- b-Frame: Body Frame

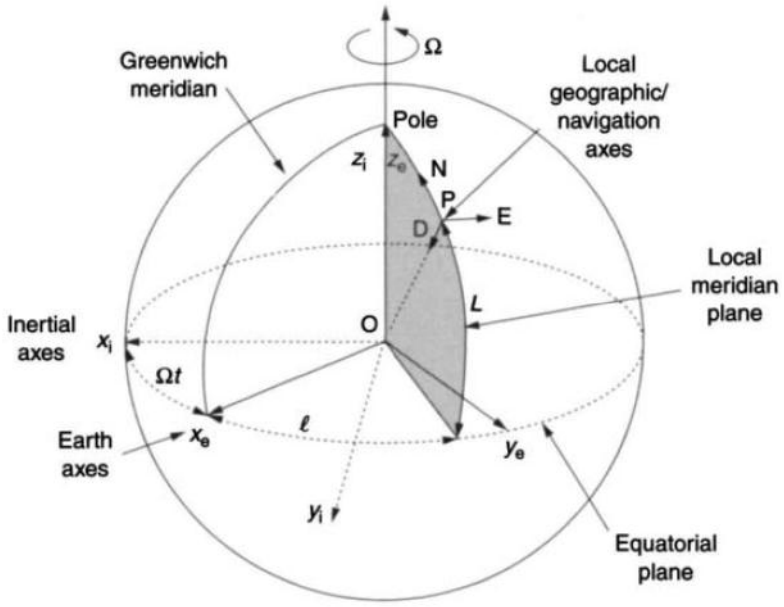

Figure 3.2 Reference Frames [4]

If navigation outputs are to be obtained in the navigation frame, the following differential equation can be used to describe the translation [4].

$$\dot{\boldsymbol{v}}^n = \boldsymbol{C}_b^n \boldsymbol{f}^b - [2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n] \times \boldsymbol{v}^n + \boldsymbol{g}^n \qquad \textbf{(3.8)}$$

where C_b^n is a direction cosine matrix used to transform any vector from body to navigation frame. This matrix can be expressed in terms of Euler angle rotations in order ψ, θ, ϕ which represent yaw, pitch, and roll respectively.

$$\boldsymbol{C_b^n} = \begin{bmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad \textbf{(3.9)}$$

$$c\theta = \cos(\theta) \quad s\theta = \sin(\theta)$$

The following differential equation is then used to describe the change in attitude described by the direction cosine matrix.

$$\dot{\boldsymbol{C}}_b^n = \boldsymbol{C}_b^n \boldsymbol{\Omega}_{nb}^b \quad \textbf{(3.10)}$$

where $\boldsymbol{\Omega}$ is the skew symmetric of the angular rate vector $\boldsymbol{\omega}_{nb}^b$. The notation below uses a cross product after the vector to denote skew symmetric.

$$\boldsymbol{\Omega}_{\mathrm{nb}}^{\mathrm{b}} = \boldsymbol{\omega}_{nb}^b \times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad \textbf{(3.11)}$$

$$\boldsymbol{\omega}_{nb}^b = \boldsymbol{\omega}_{ib}^b - \boldsymbol{C}_n^b[\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n] \quad \textbf{(3.12)}$$

The following block diagram is a representation of the navigation frame including the realistic system components required to correct the accelerometer output.

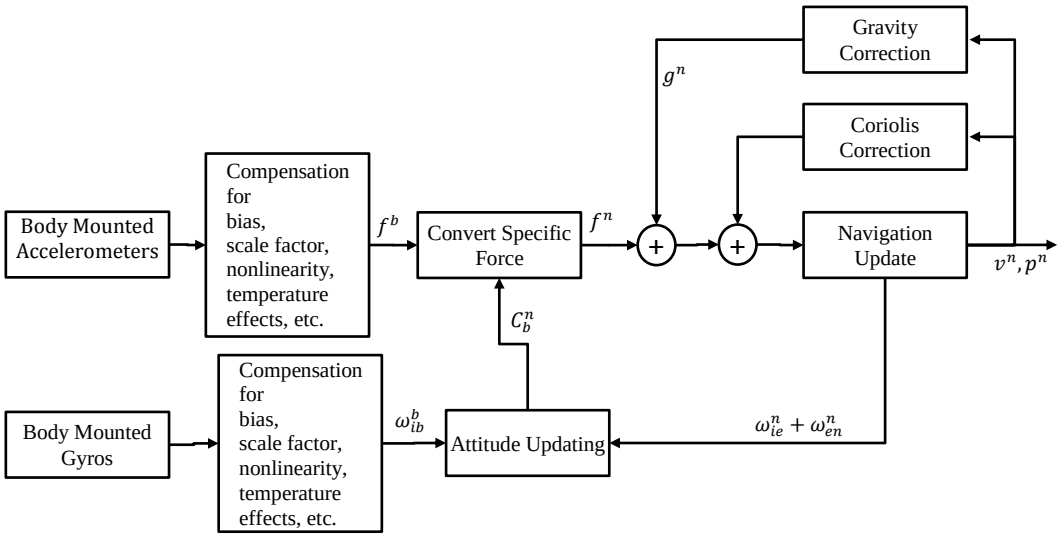

Figure 3.3 Inertial Mechanization

The majority of this thesis is devoted to the block which collects data from the accelerometers and corrects the output. The elements of this block are nearly identical to that of the gyro compensator block. The implementation issues for the mechanization will be discussed near the end of this chapter as they are relevant to the realization of an complete inertial navigation system.

3.2 Sensor and ADC Errors

The heart of the inertial measurement unit is the sensor data acquisition chain as seen in Figure 3.4. This collection of filters, converters, digitizers and integrators performs one of the key tasks of converting raw sensor measurements to digital positions and velocities.

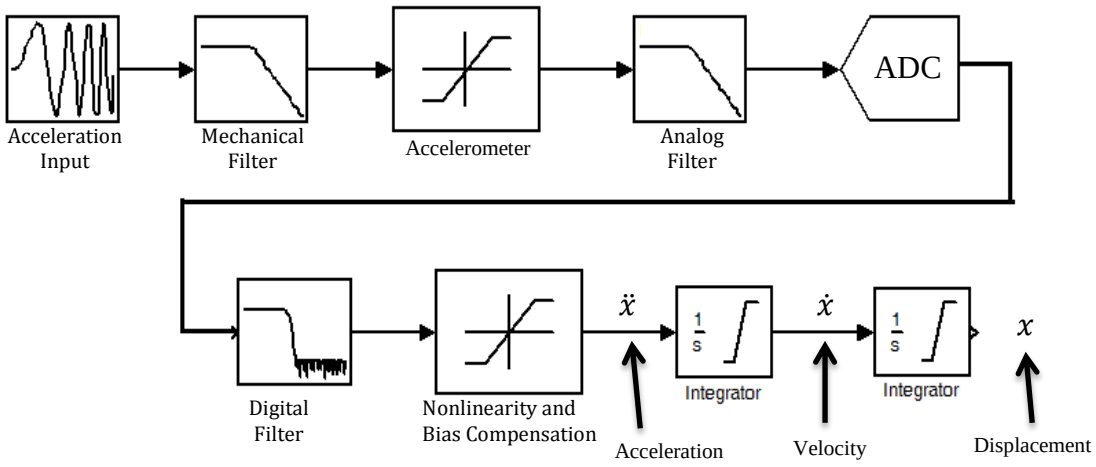

Figure 3.4 IMU Accelerometer Acquisition System Model

When considering the system design and operation in terms of minimizing error, this system must be decomposed so that each error can be isolated. Table 3.1 lists errors considered for this system. Because of the linear progression of the signal through the system, all the models are cascaded such that the errors of any particular element of the system are nearly indistinguishable from the others. Thorough system testing and identification is required to separate system behaviors. Among the key items discussed in this thesis are the errors related to nonlinearity and vibration rectification.

Table 3.1 IMU Sensor Acquisition Model

Component	Function and **System Models**	Errors
Mechanical Filter	Reduce bandwidth measured by accelerometers and gyros • **Spring Mass Damper** • **Linear and nonlinear Lowpass Filter** • **Vibration Isolation**	• Misalignment due to mass imbalance • Phase delay • Altered magnitude response • VRE and THD from nonlinear effects
Accelerometer or Gyro	Convert linear acceleration to voltage or current Convert rotation rate to voltage or current • **Mechanical filter** • **Analog filter (Electrical)** • **Servo control dynamics**	• Nonlinearity • Hysteresis • Intrinsic noise • VRE and THD from nonlinear effects • Temperature sensitive bias and scale factors • Random bias drift • Self Heating
Analog Filter (Electrical)	Reduce bandwidth of analog signal prior to ADC. Signal conditioning for ADC • **Analog filtering** • **Amplification** • **Analog Addition or Subtraction**	• Nonlinear effects THD VRE • Random bias drift • Temperature sensitive bias and scale factors • Intrinsic noise
ADC	Convert the analog voltage to digital data. • **Sample and Hold, Modulator, Demodulator, Decimator** • **Analog and digital filtering**	• Nonlinear effects THD VRE • Random bias drift • Temperature sensitive bias and scale factors • Intrinsic noise • Quantization noise • Clock jitter related conversion noise • Time synchronization errors • delays
Digital Filter	Reduce signal bandwidth, sometimes as a part of ADC conversion process. • **Digital filtering (normally Lowpass)**	• Nonlinear effects • Delays • Finite Precision Errors
Digital Compensation	Digitally compensate for known temperature dependent bias drift, scale factors, nonlinearity, alignment, cross axis sensitivity, etc.	• Nonlinear effects • Errors from inaccurate modeling or calibration
Integration or INS Mechanization	Digitally integrate the data and combine with • **Numerical Integration** • **Rotations** • **Vector and Matrix Operations**	• Integration errors causing VRE • Approximations and truncation errors cause error build up • Time synchronization errors causing VRE, sculling, coning, various other errors.

3.2.1 Bias Stability

It is standard practice when analyzing drift to perform a stationary long term bias drift test. To determine the properties of sensors considered in this thesis, 4 accelerometers were tested. Figure 3.5 shows the bias drift of these sensors whose outputs have been low-pass filtered so the drift is visible. Usually the quantization, intrinsic and cultural noise exceed the drift by 10 to 1000 times. The term "Digital" is used to describe an Applied MEMS 3 axis digital accelerometer, "Kistler" represents an 8330 capacitive force rebalanced servo accelerometer, "PCB" represents a 3701G capacitive accelerometer, and QA700 is a Honeywell Quartz servo accelerometer.

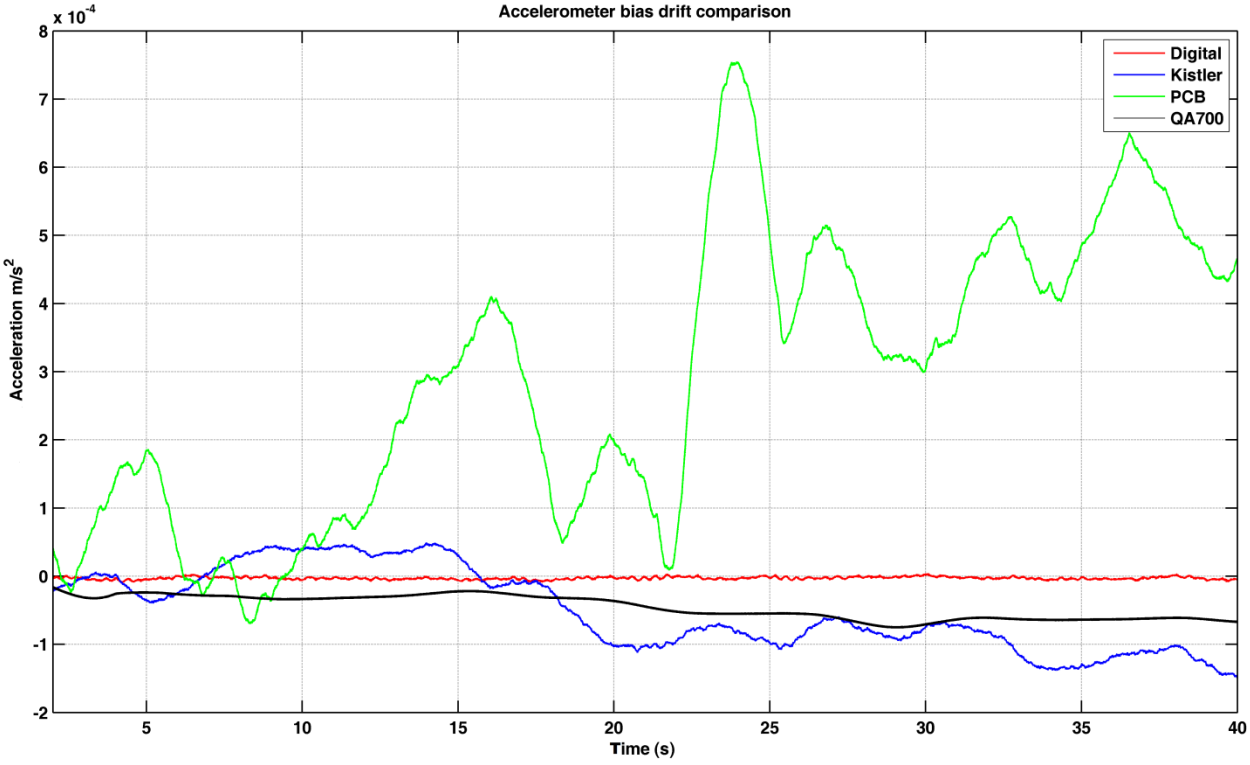

Figure 3.5 Bias Drift of Selected Accelerometers

Although it is evident that each of the accelerometers drifts, the relative quantities are difficult to compare due to the random nature of the drift. A method commonly used to characterize and compare these drifts along with other sensor characteristics is the Allan Variance. "Allan variance is a method of representing root mean square (RMS) random drift error as a function of averaging time" [1]. Figure 3.6 shows a sample Allan Variance from [14] which describes the various components of drift which can be distinguished in Allan Variance. Figure 3.7 shows the Allan variance of the 4 investigated sensors. It should be noted that the QA700 was tested at a different time of day. The resulting difference in the character of the noise added sinusoidal like oscillations in the Allan variance in frequencies corresponding to 10 to 500Hz. This is common seismic and cultural noise.

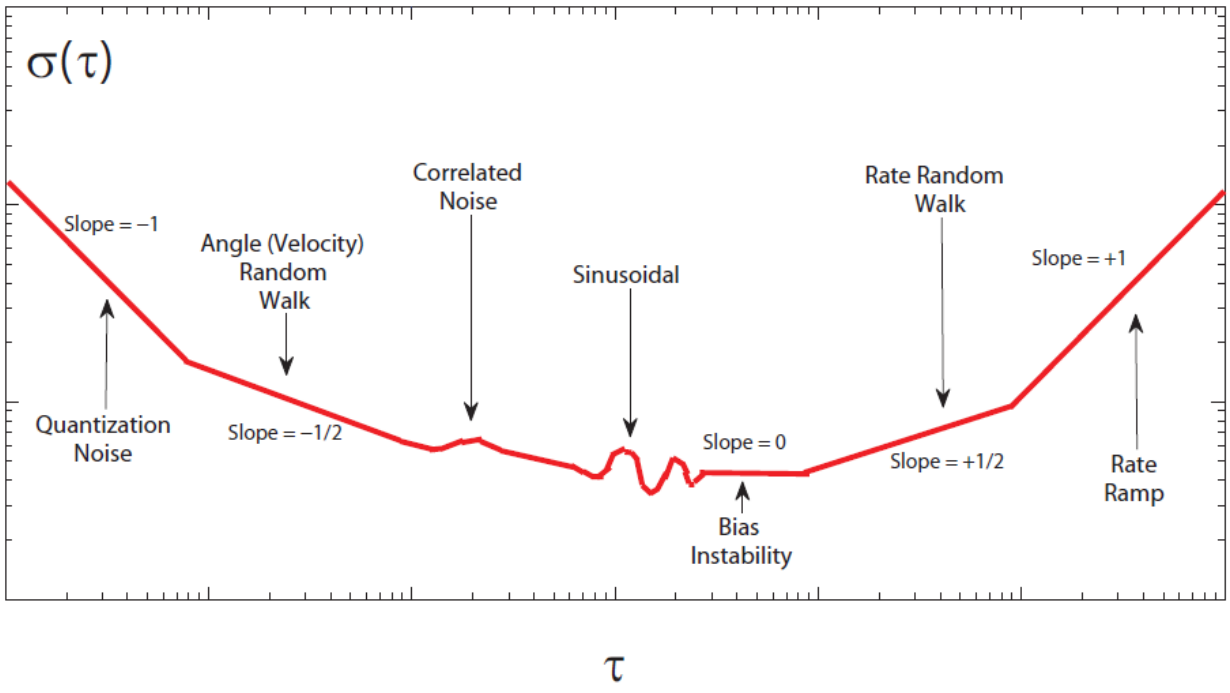

Figure 3.6 Sample Allan Variance [14]

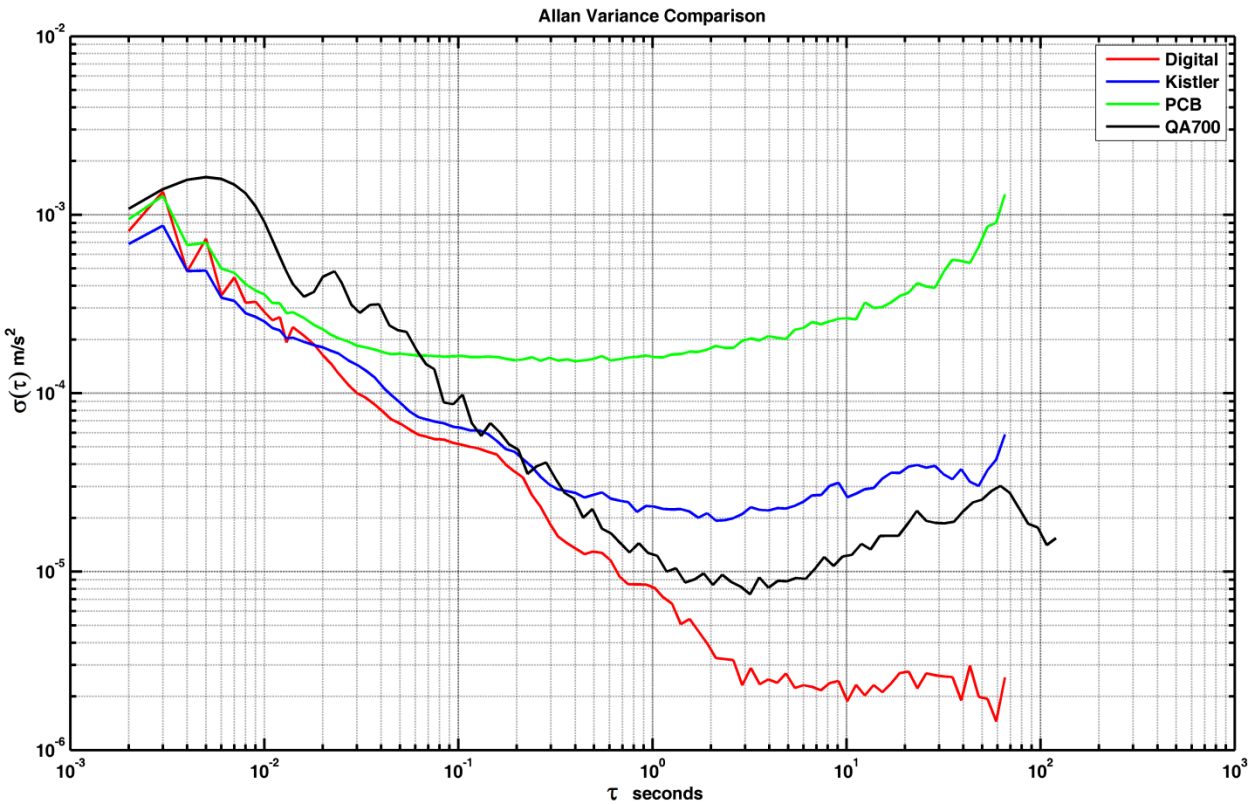

Figure 3.7 Allan Variance for Selected Sensors

With the Allan variance it is possible to determine precisely the effects of each noise contribution for a given timescale. For instance the magnitude of the random walk noise can be obtained using the following formula having evaluated $\sigma(\tau)$ and τ on the allan variance plot where the slope is $1/2$ [1].

$$K = \sigma(\tau)/\sqrt{(\tau/3)} \qquad \textbf{(3.13)}$$

where K is the random walk. For instance it is possible to estimate the expected drift for the PCB sensor over a 10 second period as $1.36 \times 10^{-4} m/s^2$. The bias instability B can also be estimated at the point where the slope of the variance is zero.

$$B \approx 1.51\, \sigma(\tau) \qquad \textbf{(3.14)}$$

Quantization noise can also be calculated using the following formula where $\sigma(\tau)\ and\ \tau$ are evaluated on the plot where the slope is -1.

$$Q = \frac{\sigma(\tau)\,\tau}{\sqrt{3}} \qquad \textbf{(3.15)}$$

The sensors tested above cost between $500 and $1500. All sensors are of the proof mass type. All of the sensors except the PCB are force rebalanced accelerometers. Of interest is the similarity of the Kistler and QA700, one of which is a MEMS where the other a "meso-scale" mechanical sensor.

3.2.2 Nonlinearity and Vibration Rectification Error

Nonlinearity is an intrinsic property of many systems such as accelerometers, gyros, electronic and mechanical components. The common causes are the nonlinearities of flexures, sensing or driving capacitors, driving coils, fluid motions, thermal changes, and other electrical elements. Nonlinearities can also be induced in accelerometers by analog or digital filters if they are incorporated in their sensors.

Nonlinearity is usually described in one of two ways according to [1], as a percent of full scale, or in terms of vibration rectification. The decision of which to use is usually based on the type of sensor and the application. It is common for MEMS sensors to be specified in terms of percent, where navigation grade sensors are given in terms of vibration rectification error from the g-squared term which has units $\mu g/g^2$.

Nonlinearities can be divided into two major types, static and dynamic. Static nonlinearities are those which are not a function of time and are thus continuously predictable as functions of measured parameters. Dynamic nonlinearities are those which

have time dependence. Examples of dynamic nonlinearities are hysteresis, and self-heating.

While many forms of nonlinearity exist, an n^{th} order polynomial is used in this derivation because of its ability to approximate a broad range of continuous static nonlinearities which account for a significant portion of total error. A discussion on the choice of order will be presented later.

The first step is to define a static nonlinearity such that the output $y(x)$ is a nonlinear function of the input x.

$$y(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + \cdots + a_nx^n \quad (3.16)$$

A sinusoidal excitation of the input is introduced where r is the magnitude of excitation, w is the frequency, and t is time.

$$x(t) = r\sin(w\,t) \quad (3.17)$$

By substituting $x(t)$ from equation (3.17) into equation (3.16) the following equation is obtained.

$$y(t) = a_0 + a_1r\sin(w\,t) + a_2r^2\sin^2(w\,t) + a_3 + r^3\sin^3(w\,t) + a_4r^4\sin^4(w\,t) + a_5r^5\sin^5(w\,t) + \cdots + a_nr^n\sin^5(wt) \quad (3.18)$$

To make the results more useful it is necessary to reduce the powers of the sine terms using either the multiple angles theorem discussed in [15] or using the trigonometric Fourier series as shown in [8]. In this derivation the latter method is used. The series expansion takes the following form.

$$y(t) = \alpha_0 + \alpha_1 \cos(wt) + \alpha_2 \cos(2wt) + \alpha_3 \cos(3wt) + \cdots \quad (3.19)$$
$$+\beta_1 \sin(wt) + \beta_2 \sin(2wt) + \beta_3 \sin(3wt) + \cdots$$

The series coefficients can be calculated using the following relations.

$$\alpha_0 = \frac{1}{T}\int_{t_0}^{t_0+T} y(t)\, dt \quad (3.20)$$

$$\alpha_n = \frac{2}{T}\int_{t_0}^{t_0+T} y(t)\cos(nwt)\, dt \quad (3.21)$$

$$\beta_n = \frac{2}{T}\int_{t_0}^{t_0+T} y(t)\sin(nwt)\, dt \quad (3.22)$$

where T is the period of oscillation, $\alpha_0$is the average value of $y(t)$, α_n and b_n are the higher order terms. The series has been truncated to show coefficients for the 5th order polynomial which are shown in the following table.

$$\alpha_0 = \frac{1}{2\pi}\int_{-\pi}^{\pi} y(t)\, dt \;=\; a_0 + \frac{1}{2}a_2 r^2 + \frac{3}{8}a_4 r^4 \quad (3.23)$$

$$\beta_1 = \frac{1}{\pi}\int_{-\pi}^{\pi} y(t)\sin(wt)\, dt \;=\; a_1 r + \frac{3}{4}a_3 r^3 + \frac{5}{8}a_5 r^5 \quad (3.24)$$

Table 3.2 Trigonometric Fourier Series Coefficients for 5th Order Polynomial

n	α_n	β_n
0	$a_0 + \frac{1}{2}a_2 r^2 + \frac{3}{8}a_4 r^4$	
1	0	$a_1 r + \frac{3}{4}a_3 r^3 + \frac{5}{8}a_5 r^5$
2	$-\frac{1}{2}(a_2 r^2 + a_4 r^4)$	0
3	0	$-\frac{1}{16}(4a_3 r^3 + 5a_5 r^5)$
4	$\frac{1}{8}a_4 r^4$	0
5	0	$\frac{1}{16}a_5 r^5$

The resulting equation yields.

$$y(t) = a_0 + \frac{1}{2}a_2 r^2 + \frac{3}{8}a_4 r^4 \quad (3.25\text{ a})$$

$$+\left(a_1 r + \frac{3}{4}a_3 r^3 + \frac{5}{8}a_5 r^5\right)\sin(wt) \quad (3.25\text{ b})$$

$$-\frac{1}{2}(a_2 r^2 + a_4 r^4)\cos(2wt) \quad (3.25\text{ c})$$

$$-\frac{1}{16}(4a_3 r^3 + 5a_5 r^5)\sin(3wt) \quad (3.25\text{ d})$$

$$+\frac{1}{8}a_4 r^4 \cos(4wt) \quad (3.25\text{ e})$$

$$+\frac{1}{16}a_5 r^5 \sin(5wt) \quad (3.25\text{ f})$$

From the form of this equation it is possible now to identify both the constant vibration rectification error as seen by the constant terms in equation (3.25 a) and the harmonic distortion terms described by equation (3.25 c-f). The original frequency is also distorted in amplitude as seen by extra the terms found in equation (3.25 b).

It is the constant error response that explains the definition of vibration rectification. "A Steady-State error in the output while vibratory disturbances are acting on an accelerometer" [16]. Although vibration rectification is defined for accelerometers, they are by no means the only sensors affected. In theory all sensors or systems with nonlinearities can suffer from some form of vibration rectification.

Equation (3.25) also leads to the conclusion that harmonic distortion is a result of the same nonlinearity. To illustrate this with a simple example, a 5th order polynomial nonlinearity is introduced into a signal as shown in Figure 3.8. This system is then excited with a sinusoid and the resulting frequency response is plotted. It is not surprising that the harmonic distortion from this nonlinearity is very large, as the nonlinearity in this example

represents about 15% of full scale. By comparison a medium grade accelerometer will have between 1% and 0.1%. A navigation grade accelerometer will have less than 0.01%. It should also be clear why it contains only 5 harmonics. The number of harmonics is equal to the order of the nonlinearity. It is possible to quickly determine the order of the nonlinearity by applying a constant sinusoidal excitation and counting the number of visible peaks. However in real signals it is possible that these harmonics will be hidden in noise. This noise practically limits the ability to identify and correct nonlinearity.

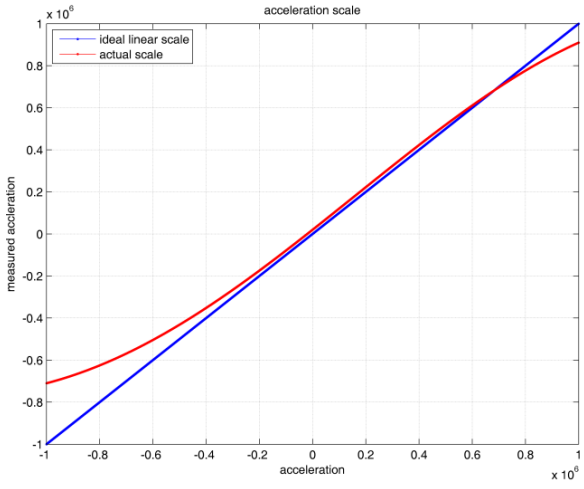

Figure 3.8 Sample Nonlinearity

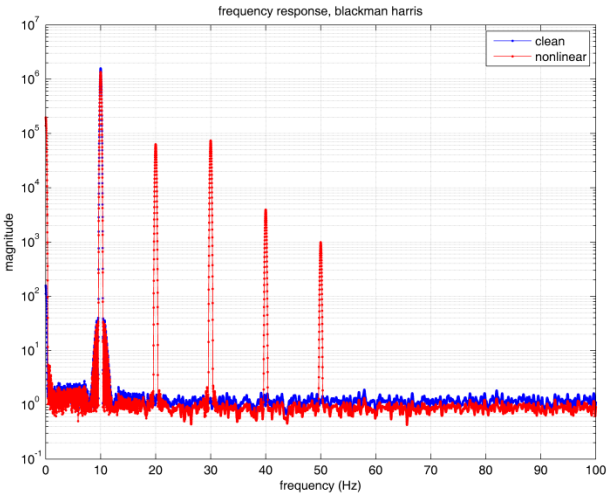

Figure 3.9 Sample Frequency Response

Various sources of error for the Q-Flex accelerometer are identified in Jacobs [9]. Among the first considered are possible nonlinearities from the deflection of the proof mass on the pendulous beam. The angle of deflection is related to the servo stiffness which is a function of the capacitive displacement sensor and the strength and response time of the servo control. By minimizing this displacement the effects of nonlinearity and cross axis sensitivity are minimized. It is stated that the angle of displacement in this sensor is limited to $5 \times 10^{-6}\ radians/g$. Other error sources are identified such as the nonlinearity of the rebalance servo.

In each case described above, the design of the accelerometer has been modified to compensate for these errors. The residual error is reflected by a reasonably low nonlinearity seen in the Q-Flex Accelerometers.

The nonlinear performance of 3 of the tested sensors is shown in the following table for the same 1 g range. MEMS and mechanical sensors are specified differently. As a result, this data is a mix of specifications from the data sheets [17] [18] [19] and actual testing. The excellent performance of the QA700 is contrasted by the relatively poor performance of the MEMS sensors. Recall that these sensors all fall within a relatively close price range around $500 to $1500. These results indicate the importance of nonlinear compensation when under vibration, especially for MEMS accelerometers with high nonlinearity.

Table 3.3 Nonlinearity and VRE of Tested Sensors

Sensor	Non-linearity [%]	VRE Tested from 0 to 1 g $[ug]$	VRE Tested from 0 to 1g Post Compensation $[ug]$	1 minute displacement error Pre/post Correction
QA700	0.001^1	$\approx 0\ to 10^1 \quad 50^2$	$\approx 0\ to\ 0.1^1$	17 cm / 1.7 mm
PCB	1^2	$\approx 0\ to\ 2000^1$	$\approx 0\ to\ 20^1$	35 m / 0.35 m
Kistler	0.2^2	$\approx +1500\ to -500^1$	$\approx +15\ to -5^1$	26 m / 0.26 m

[1] Experimentally Tested [2]From Data sheet

3.2.3 Vibration Rectification vs Amplitude

While it is convenient to think of VRE as a constant, it is apparent from equation (3.25 a) that it is in fact a relation of the vibration amplitude. This becomes a major problem as the vibration amplitude in many situations is not constant nor is it of a single frequency. As the vibrations change the magnitude of the vibration rectification does as well. This problem

alone prevents the use of Kalman filters to track the bias drift due to VRE unless they have been specifically designed to model these errors.

3.2.4 Vibration Rectification vs Frequency

Due to the complex interactions of mechanical, electrical and fluid elements inside an accelerometer, a frequency dependent component of nonlinearity exists. The physical and electrical dynamics of every accelerometer limit the bandwidth in much the same way as a filter. In a servo accelerometer the rebalance electronics mostly compensate for these dynamics up to certain frequency, leaving a mostly flat frequency response for lower frequencies [9]. These dynamics are not as well compensated in non-servo accelerometers which do not limit the movement of the proof mass.

The variation of damping characteristics over different frequency ranges causes a variation in the nonlinearity and thus the vibration rectification. An example of this is seen in the datasheet of a QA700 accelerometer [17] which specifies $< 50\mu g/g^2 rms$ at 50-200Hz and $< 150\mu g/g^2 rms$ for the range 750-2000Hz.

3.2.5 Self-Heating and Temperature Sensitivity

The temperature sensitivity of many sensor parameters is well known. This is particularly well understood for temperature dependent bias which can vary from 10 to $1000\mu g/$°C. The method of compensation for this error requires precision temperature measurement. By measuring the temperature and applying compensation, the temperature dependent drift can be removed.

Power dissipation of various components in the sensor results in heat generation. The quiescent power dissipation will result in a constant elevated temperature. This

temperature can easily be monitored by temperature sensors placed inside or even outside of the sensor. Dynamic power dissipation will cause additional thermal gradients within the sensor whose effects are not measured until a considerable time has passed.

Dynamic self-heating is a greater concern when using force rebalanced accelerometers, particularly the larger mechanical accelerometers which require more power. This is due to the variable level of power dissipation required to hold the proof mass. If the temperature sensor could measure directly on the sensitive element the problem would be easily managed. Depending on the distance between the elements and the thermal resistance between them a delay of seconds to minutes could result. Under highly variable conditions or rapidly changing vibrations, the sensitive element can be elevated 0.1 to 1°C from the measured temperature which results in an uncompensated bias error . For this reason it is recommended to model the power dissipation and effectively estimate the temperature well before it is sensed. IEEE Std 836-1991 [20] recommends the following equation to model the temperature rise.

$$\Delta T(t) = \rho P(a)\left(1 - e^{-\frac{t}{\tau}}\right) \qquad (3.26)$$

where:

ΔT = temperature rise at a critical location of accelerometer

$P(a)$ = power dissipation as a function of acceleration

ρ = thermal resistance to the heat sink

τ = thermal time constant

t = time since onset of acceleration

The best solution is to perform this calculation in real time on the continuously changing signal adding it to the measured temperature.

Of the sensors being tested only two are equipped with onboard temperature sensors. The QA700 outputs a current proportional to temperature which can be measured with an ADC to a resolution better than 10^{-4} °C. The Digital accelerometer has a digital temperature sensor with a 1°C resolution. The other MEMS sensors are not equipped with internal temperature sensors. The following table lists the temperature sensitivities of the tested sensors as given in their datasheets [21] [17] [19] [18] [22]. The QA2000 was added to this list to demonstrate the similarity to the QA700.

Table 3.4 Temperature Sensitivities of Selected Sensors

Sensor	Bias Sensitivity [μg/°C]	Scale-Factor Sensitivity [ppm/°C]	Displacement error due to 1°C measurement error for 1 minute [meters]
QA700	70	200	1.23
PCB	5606	910	99
Kistler	200	100	3.53
Digital	100	75	1.76
QA2000	30	180	0.529

Based on these results the PCB again seems to be a poor choice for a navigation instrument. Even small temperature measurement errors will result in large errors. The other sensors seem to have similar sensitivities.

3.2.6 ADC Noise Analysis

During the process of system design and analysis one of the key tasks is to determine the performance of the ADC as operated in the system. ADCs and their supporting circuitry are responsible for adding further noise to the system, while the clock jitter is usually a major factor in the noise performance of the ADC [23]. Other noise sources include "1/f" noise,

shot noise, Shottky noise, resistor noise, and external factors such as electromagnetic and conducted interference. The following analysis are performed both to determine the required specifications and design parameters for the hardware and to determine the performance once operational.

3.2.6.1 Clock Jitter and ENOB

The signal to noise ratio SNR and effective number of bits ENOB[1] are used to describe the true performance of the ADC. One of the major contributors to SNR loss is the clock jitter. According to [24] jitter is a variation in the placement of a clock edge. This error can be represented by various types of jitter such as period jitter, phase jitter, duty cycle jitter and others. These timing errors cause direct errors in the amplitude and frequency of instantaneous ADC conversions. A relationship exists to determine the signal to noise ratio of the ADC for a given sample rate and jitter. The following equation defines the SNR (dB) for an ideal ADC a with clock jitter of period t_j operated at frequency f.

$$SNR_{ideal}(f, t_j) = 20 \log_{10} \left[\frac{1}{2 \pi f t_j} \right] \quad (3.27)$$

For comparison, the SNR of a jitter free ADC with N bits of resolution is found using the following equation.

$$SNR_{bit}(N) = [6.02N + 1.76] \quad (3.28)$$

Figure 3.10 combines these equations to determine the effective number of bits from a given oscillator jitter and sampling frequency.

---

[1] ENOB Effective number of bits. Specifies the number of bits in the digitized signal above the noise floor.

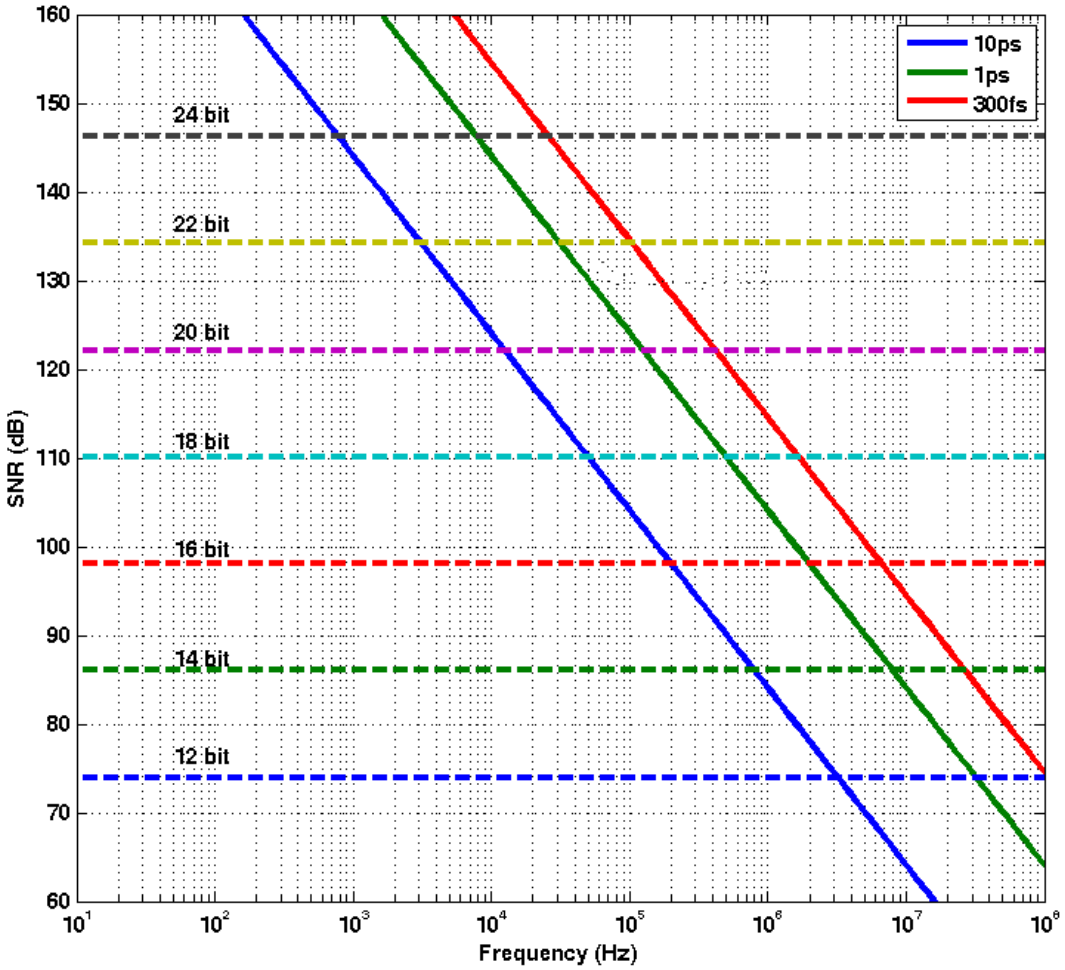

Figure 3.10 SNR and ENOB Due to Clock Jitter

According to this figure, if a 24bit ADC is used at 50kHz with a clock that has a 1 ps jitter, the effective resolution will be closer to 21 bits.

To determine if an improvement was indeed possible with the ADC used in this testing, a shorted input test was conducted with the original oscillator and with a lower jitter oscillator featuring 1 ps jitter. The results in Figure 3.11 show a clear reduction in higher order harmonics present in the spectrum.

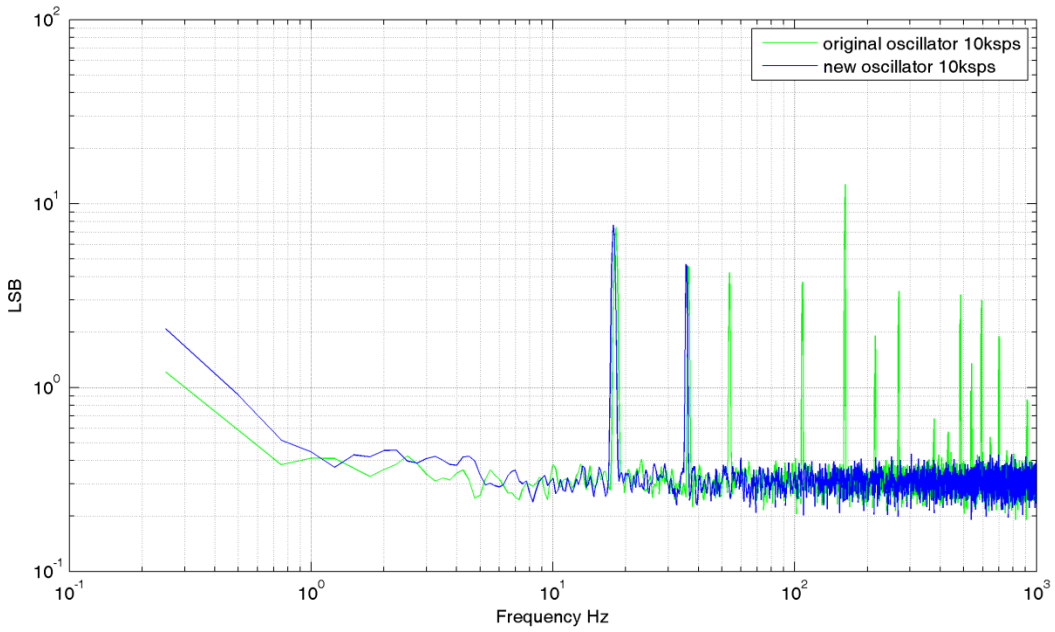

Figure 3.11 Noise Test for Clock Jitter Analysis

3.2.6.2 Input Noise

The noise floor of the ADC is determined by a number of intrinsic factors but the overall noise is also affected by the front end amplifier noise. It is necessary to choose an amplifier which does not significantly increase noise. Moghimi recommends using the following equation to determine the net SNR degradation [23].

$$SNR_{loss} = 20\log_{10}\left(\frac{V_n}{\sqrt{V_n^2+\frac{\pi}{2}f\left(\frac{2.5\,G\,e_n}{FSR}\right)^2}}\right) \quad (3.29)$$

where:

V_n rms noise of ADC in μV

f -3 dB input bandwidth of ADC in MHz

G amplifier gain

e_n equivalent input noise voltage spectral density of amplifier in $nV/\sqrt{Hz}$

FSR full scale input span of ADC in V

Using this equation, the SNR degradation can be found for various amplifier noise specifications. The following values are obtained from the ADS1278 datasheet [25].

$V_n = 5.5\ \mu V \quad f = 0.49 * f_{sampling} = 0.49 * 50kHz \cong 0.025MHz$ FSR=5V

The first two amplifiers shown in Table 3.5 are the two fully differential[1] amplifiers recommended for this ADC. The second two amplifiers are used for buffering the accelerometer signal at the source. The operational specification of the parts are similar except that the THS4521 operates at 1 mA quiescent current where the OPA1632 operates at 14 mA.

Table 3.5 SNR Reduction Due to Input Amplifier Noise

Amplfier	$e_n\ (nV/\sqrt{Hz})$	SNR_{loss} (dB)	I_q mA
OPA1632	1.3	0.002	14
THS4521	4.6	0.02	1
LT1012	14	0.27	0.4
OPA211	1.1	0.002	3.6

It would seem that these numbers indicate that the amplifier does not significantly impact the noise of the system. This is the case in this system given the relatively low gain of 1.0, however if a higher gain is required or if multiple amplifiers are cascaded these errors become more significant. While the choice may seem obvious from a noise analysis point of view to pick the lowest noise part, the reduction in power consumption over 16 channels using the THS4521 vs the OPA1632 is over 6 watts. This power would generate a

[1] Fully differential amplifiers have differential inputs and outputs as opposed to single output.

significant amount of heat which would contribute to thermal drift throughout the system as well as draw more current which could cause more power supply noise. A similar tradeoff is present in selecting between the buffer amps. While the lower noise is better, the proximity of the amplifiers to the sensors can directly conduct heat to the sensors causing thermal drift.

3.2.6.3 Thermal Noise

Thermal noise, a.k.a. johnson noise or resistor noise, represents a source of white noise which occurs in resistors. A simple relationship exists to calculate the rms voltage noise of a given resistor [26].

$$E_{th} = \sqrt{4kTRB} \qquad (3.30)$$

where:

E_{th} Thermal noise voltage (V rms)

k Boltzmann's constant (1.38×10^{-23})

T Absolute temperature (Kelvin)

R Resistance (ohms)

B Noise bandwidth Hz ($f_H - f_L$)

It is clear from this relationship that reducing resistor values is important to reducing thermal noise. There is however a tradeoff as mentioned by [26], in that reducing the power consumption by scaling up the resistors may reduce 1/f noise and related drift. While it is important for an IMU acquisition system to have low noise at high frequencies, it is also necessary to minimize long term drift. Further analysis and testing will be required to determine the optimal configuration.

3.2.6.4 "1/f" and White noise

The noise components listed above are useful for determining the noise sources for design purposes. The following analysis is used to determine the overall performance based on noise testing. White noise, also known as broad band noise, represents the flat part of the noise spectrum. The "1/f" noise, also called pink noise, goes up inversely proportional to frequency. A simple relation exists to determine the total rms voltage noise E_n as a function of the white noise voltage e_n, corner frequency f_{nc}, and the bandwidth defined by f_L and f_H [26].

$$E_n = e_n\sqrt{f_{nc}\ln\frac{f_H}{f_L} + (f_H - f_L)} \qquad (3.31)$$

The noise spectrum shown in Figure 3.12 represents the results obtained while shorting the input pre-amplifier of the ADC.

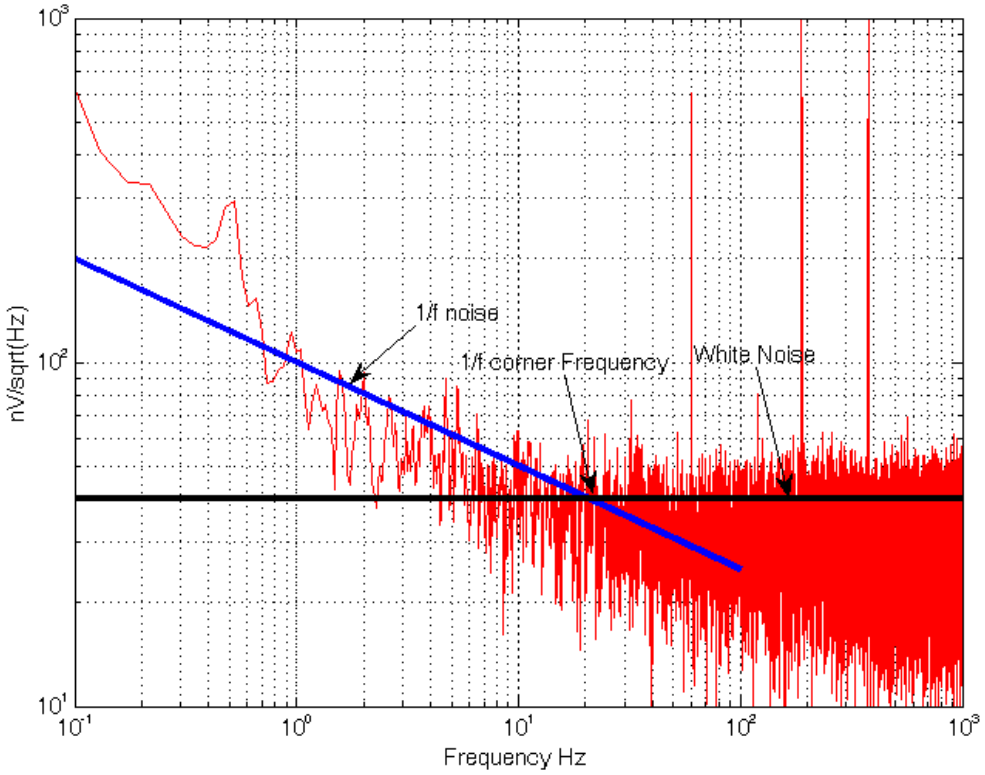

Figure 3.12 Noise Spectrum in the ADC with Shorted Amplifier Inputs

Given a $35nV/\sqrt{Hz}$ noise floor and a corner frequency of 20Hz, the noise can be calculated for a selection of bandwidths. Given that DC terms are required, a lower band frequency of 0.001 Hz is kept in all cases.

Table 3.6 ADC Voltage Noise Analysis

f_H	E_n (μV)
50000	7.85
10000	3.56
1000	1.25
100	0.636
10	0.487
1	0.412

This analysis is supported by the specifications in the data sheet for the ADS1278 ADC which states 5.5 μV of noise in high resolution mode at 50kHz bandwidth. The extra noise is likely a result of the amplifiers and supporting components which are not present in the datasheet specifications for the ADC. It should also be noted that the noise spectrum in Figure 3.12 at frequencies below 1Hz is higher than the theoretical 1/f noise. This can be attributed to thermal and other drift terms. To verify the results, the Allan Variance of the noise test is generated. Figure 3.13 shows that about 6uV of noise is present at the highest frequency (left side of Allan Variance).

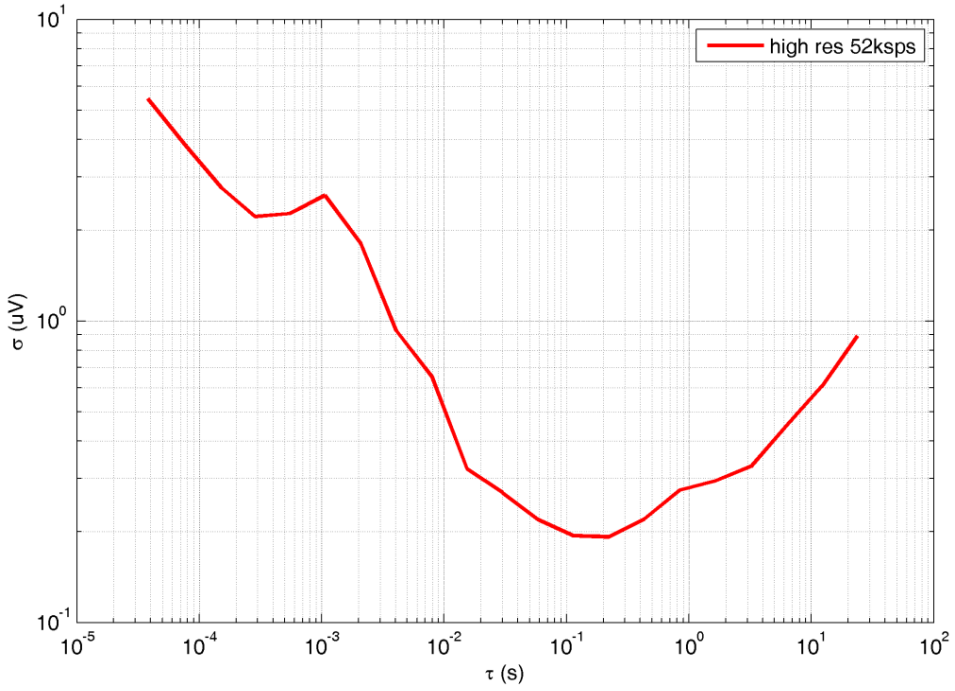

Figure 3.13 ADS1278evm Allan Variance, Shorted Amplifer Input

3.2.7 Delay Errors

A delay is present in most dynamic systems, including mechanical and electrical filters, digital data paths, ADCs, digital filters, etc. It can also be generated simply by a lack of precise timing as one signal is compared to another or against a reference clock. This delay can cause serious errors for high vibration environments or in situations where the dynamics are at high frequencies. In this research the two major issues are the differences in delays of different sensors and, during calibration, delays between the reference sensor and the tested sensor.

In navigation systems where accelerometers and gyros are combined, delay effects can cause errors between the changes in attitude and the acceleration. This can result in "artificial" coning and sculling which, when interacting with the mechanization can cause attitude and position drift.

The following equation describes the normalized error[1] E for a given delay t_d and a given frequency of oscillation f. Figure 3.14 plots error for a selection of oscillation frequencies for delays between 1 us and 1 ms.

$$E = 1 - \cos(2\pi f t_d) \qquad (3.32)$$

[1] Dimensionless

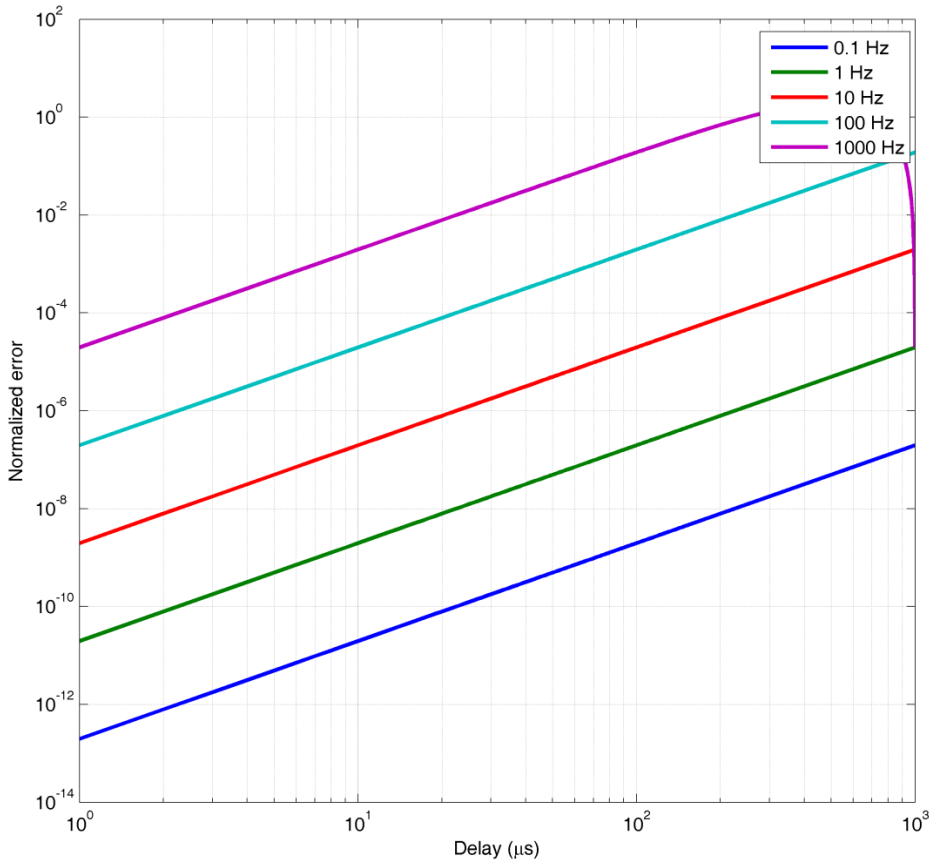

Figure 3.14 Normalized Delay Errors

It is clear that as the frequency of oscillation increases the error increases. For example, with a 1 V sinusoidal oscillation at 1 Hz, 1 μV accuracy can be obtained with delays as large as 200 μs. However for 10 Hz, a maximum 20 μs delay is needed. This is even more critical with higher frequencies such as 100 Hz where only 2 μs is allowable.

3.2.8 Sampling and Integration Errors

Inertial measurement units are used to measure accelerations and rotation rates but in most cases these signals are integrated to obtain position, velocity and attitude. Therefore an important part of the inertial system is the integration process. In the continuous domain the integration of a signal regardless of its content will be exact. This is not the case for the discrete domain. Errors in numerical integration can cause unpredictable

position, velocity and attitude shifts which can grow over time. This error in integration can cause a type of vibration rectification where a continuous average of the errors is not zeros mean. This type of error is not predictable. The best solution is to improve the accuracy of integration. While higher order approximations are available an alternate solution is to decrease the step size of the integration. This solution not only improves the approximation of the integral, but it also reduces the time delay and synchronization errors which can cause a number of errors in the INS mechanization. These errors are described in [5].

3.3 Analysis and Compensation of Errors

To demonstrate some of the errors described above and develop correction algorithms an accelerometer simulator was developed. Using a simulator it is possible to isolate types of errors in a manner not usually possible in real systems. The compensation methods can be developed and verified before testing on real systems.

3.3.1 Vibration Rectification Due To Nonlinearity

The following example is a demonstration of vibration rectification error caused by nonlinearity in the accelerometer. It is also an example of Least Squares estimation method being used to estimate the nonlinear model of a simulated burst series. A series of 3 bursts with amplitude $12.5 m/s^2$ of 1 second length are applied to a simulated accelerometer with nonlinearity and bias drift. The nonlinearity of the simulated accelerometer can be described by a polynomial with coefficients shown in Table 3.7.

Table 3.7 Coefficients of Sample Nonlinearity

Coefficients	Values
a_0	0.0
a_1	1.0
a_2	1.642e-5
a_3	1.3665e-6
a_4	-1.286e-7
a_5	-1.17e-8

The plots are presented in velocity as the bias and VRE errors are more easily noticed. At this scale Figure 3.15 shows little error in drift or VRE as the magnitude of the errors is too small to see.

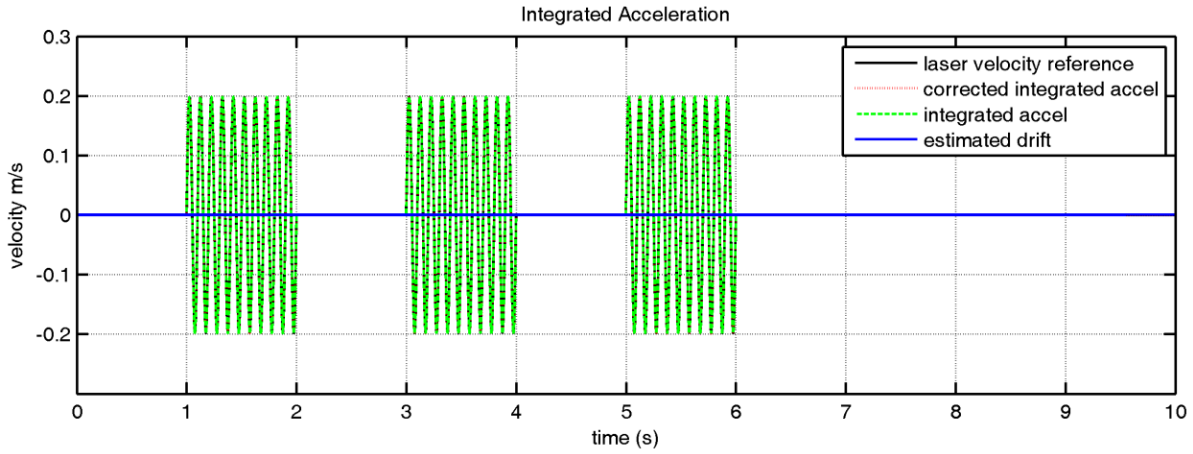

Figure 3.15 Burst Test Simulation

By subtracting the estimated drift from the original integrated acceleration the VRE is now easily distinguished, as shown in Figure 3.16, and can be calculated directly. As stated before, the definition of VRE is an apparent bias shift in acceleration due to oscillation. As this is a zero mean excitation, the expectation is that when no excitation is present, no velocity change is seen, which is supported by the example. It should also be clear that

during excitation the bias shift in acceleration causes a change in velocity which is proportional to the length of excitation.

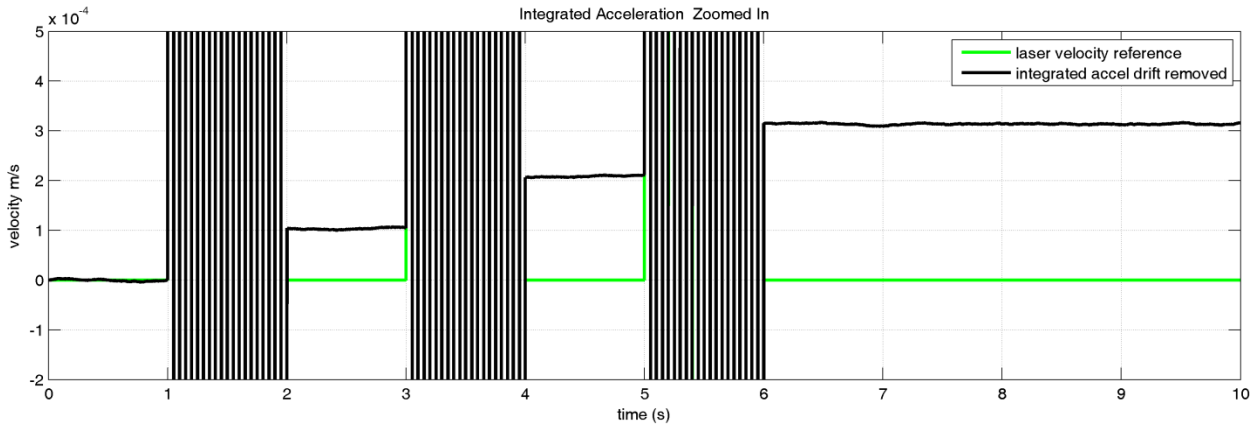

Figure 3.16 Burst Test Simulation VRE

To verify that the change in velocity is only occurring during oscillation a different part of the test is shown. Figure 3.17 zooms in on the bottom of the oscillations where the envelope of the oscillations for the reference and the integrated acceleration are visible. It can be seen that the velocity changes steadily under excitation. Reference lines have been placed on the figure to show that there is negligible drift between bursts as opposed to the steady change during. VRE can be computed as the slope of the change during excitation.

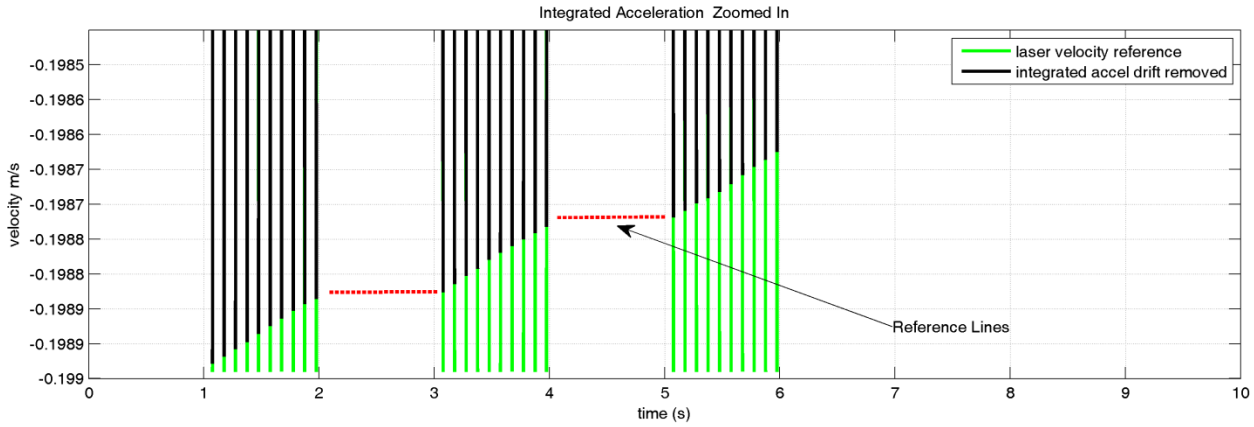

Figure 3.17 Burst Test Simulation VRE

The magnitude of the expected VRE can be calculated using equation (3.25 a), rewritten below

$$VRE = a_0 + \frac{1}{2}a_2 r^2 + \frac{3}{8}a_4 r^4 \qquad (3.33)$$

Using this equation the expected VRE can be computed for r=12.5 as $1.054 \times 10^{-4} m/s^2$ which is also shown in Figure 3.16. The above equation leads to the conclusion that the VRE is dependent on the vibration amplitude which, when plotted for this simulated sensor, produces Figure 3.18. It is apparent that the bias changes with varying amplitude which is why VRE cannot be resolved by applying a constant bias adjustment, but only by correcting the nonlinearity.

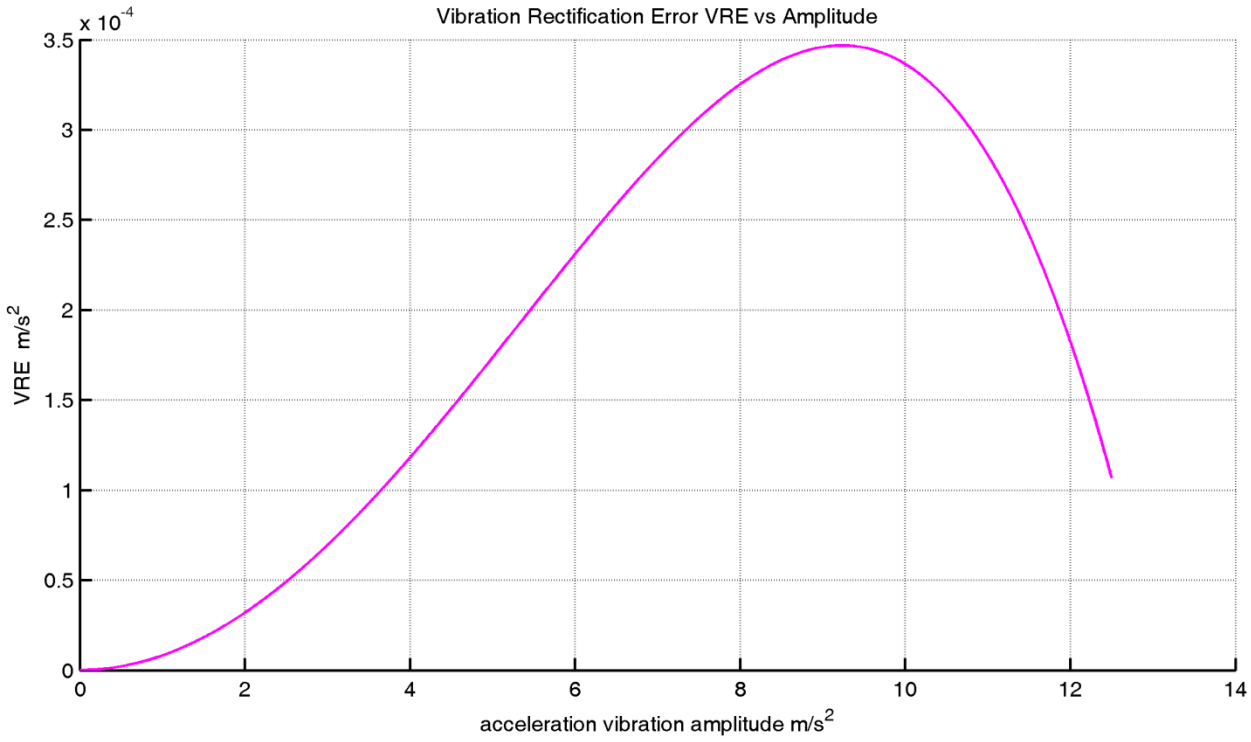

Figure 3.18 VRE vs Amplitude Simulated Data

3.3.2 Computational Vibration Rectification from Integration

The following example will demonstrate the problems of computational VRE using experimental data from a Kistler 8330 capacitive servo MEMS accelerometer. The test conducted was a 10 second triple burst test at 1 g amplitude shown in Figure 3.19. The data was sampled at 50 kHz, 1 kHz, 100 Hz and 50 Hz. The burst frequency is 10 Hz.

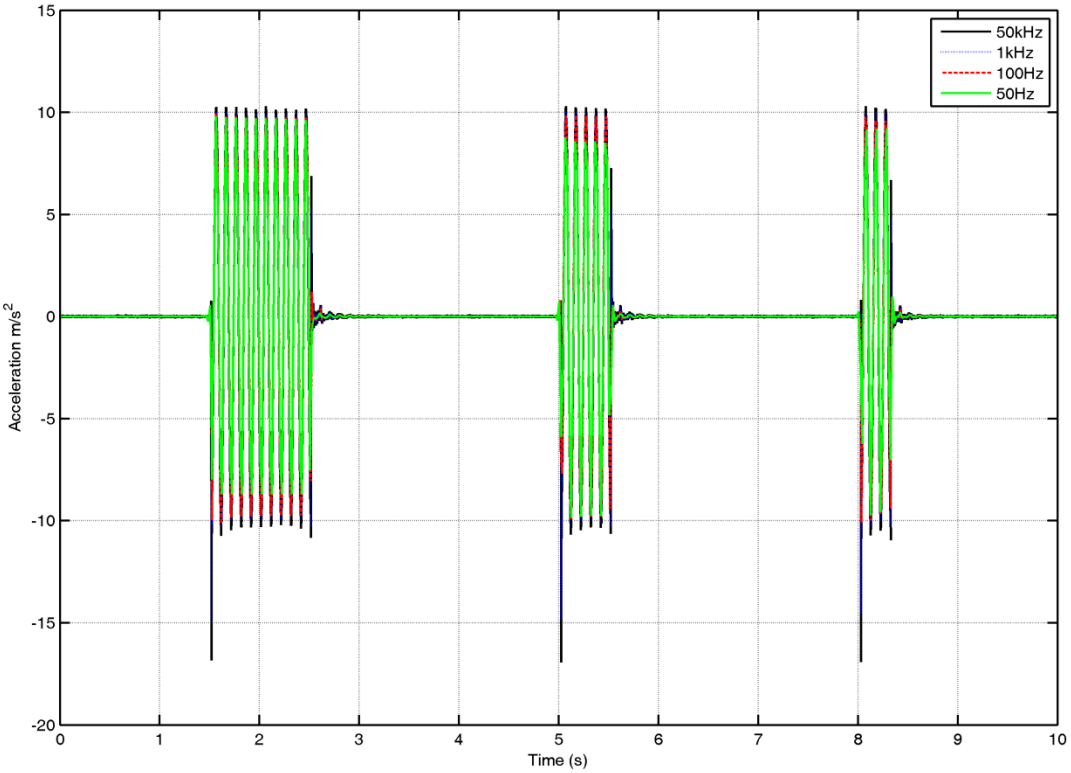

Figure 3.19 Burst Test with Kistler, Sample rates from 50 Hz to 50 kHz

It is common practice in signal analysis to choose a sample rate to be double that of the frequency of the signal to prevent aliasing. In the example here the lowest sampling frequency is 5 times the burst frequency. However that is not to say it is a good representation of the signal. Figure 3.20 shows the bottom of a single sinusoid to show the degree to which each of the sample rates was able to describe the signal. The high

frequency vibration shown only on the 50 kHz signal is a result of the stage drive noise and cultural noise.

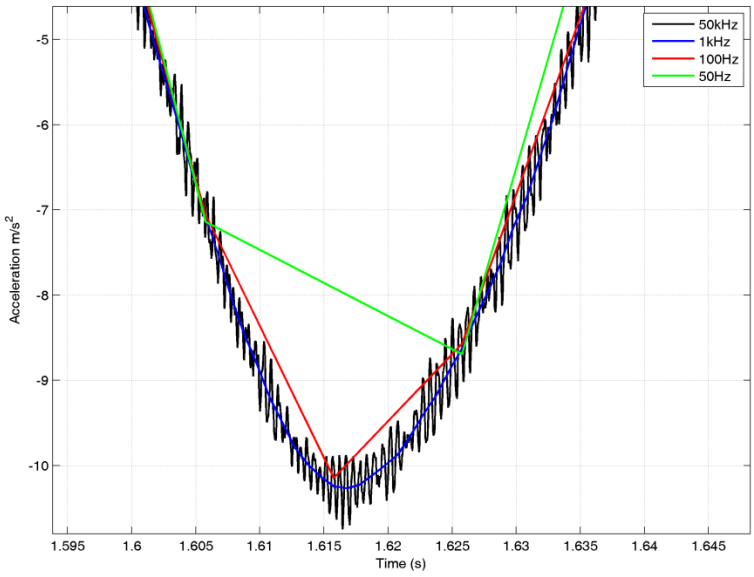

Figure 3.20 Burst Test Kistler Zoomed In

For the next part of the example the best available estimate of the inverse nonlinearity is used to correct the sensor output. The results of the integrated acceleration are shown in Figure 3.21. It is apparent that the 50 kHz and 1 kHz data have almost negligible vibration rectification. The 100 Hz and 50 Hz signals on the other hand were not fully corrected. The remaining error is due to the computational VRE introduced by errors in the integration caused by insufficient sample rate. This error cannot be resolved or predicted and will therefore be perceived as system drift or noise. The only way to avoid this is to sample at frequencies well above those seen. Experimental tests or simulations can be used to predict performance.

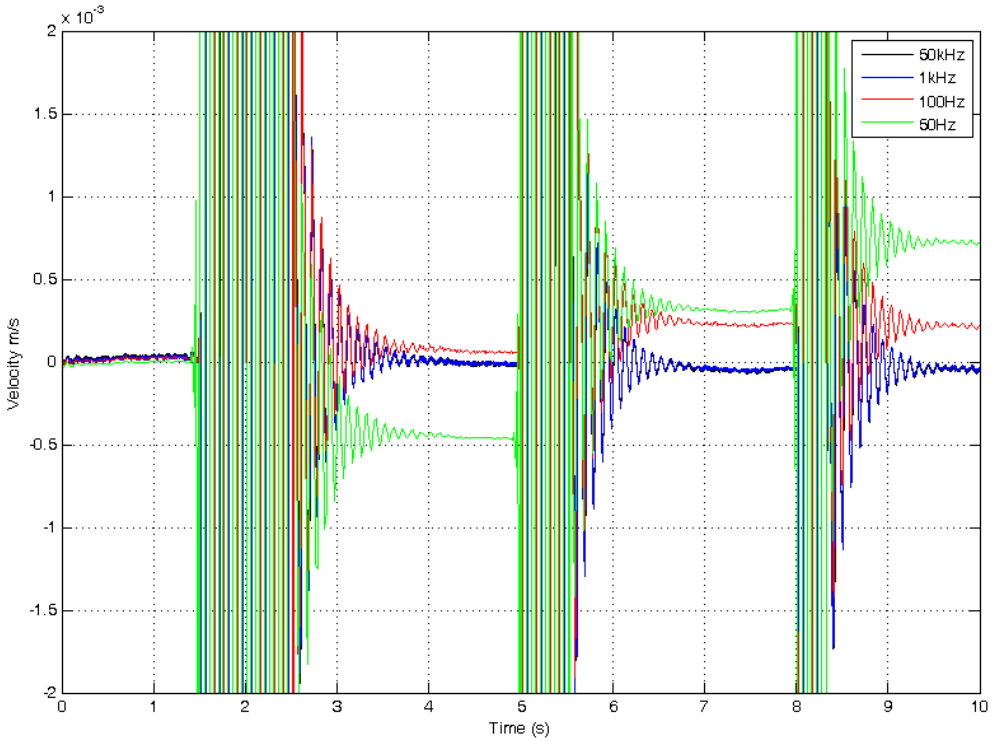

Figure 3.21 Burst Test Kistler Mult Rate Integrated Acceleration Corrected

3.3.3 Compensation of vibration rectification caused by nonlinearity

Inertial measurement units are generally designed for a certain bandwidth of dynamics. In aircraft navigation, depending on the type and size of airplane, the dynamics are slow. These dynamics can be represented by a bandwidth from DC to about 20 Hz. There are however vibrations from the engine which can be detected by the accelerometer at higher frequencies as well as impact shocks during landings which have very high bandwidth. It is common place to use a combination of mechanical (spring-mass-damper) filters, analog (electrical) filters, and digital filters to remove these vibrations prior to any mathematical operations. The following analysis shows potential problems with some implementations of these filters.

In this example, the nonlinearity $y(x)$ is defined as a polynomial with coefficients $a_0 \ldots a_5$. This polynomial is not usually known nor can it be perfectly represented. It can however be

approximated by a finite order polynomial for the purposes of estimating VRE. The inverse polynomial $\hat{x}(y)$ which is found by calibration has coefficients $c_0 \ldots c_5$. Compensating the sensor nonlinearity is simply a matter of applying the inverse non-linearity to the output signal to effectively reconstruct the input without distortion as is shown in Figure 3.22.

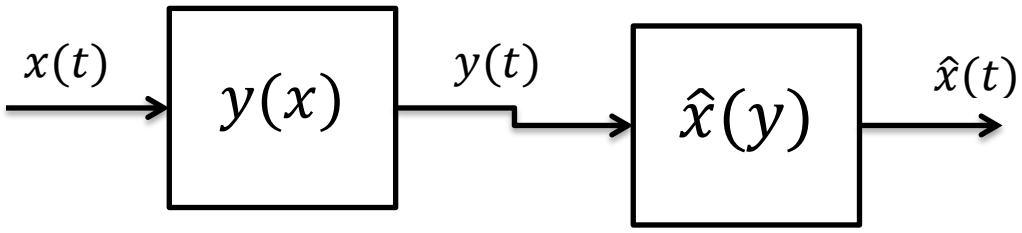

Figure 3.22 Compensation Block Diagram

3.3.3.1 Ideal Case, No Filtering

To demonstrate the best case in compensation the following example is presented. The coefficients of the nonlinearity and it's inverse are shown in Table 3.8 and Table 3.9.

Table 3.8 Polynomial Coefficients of Sensor Nonlinearity

Coefficients	Values
a_0	0.0
a_1	1.0
a_2	1.642e-5
a_3	1.3665e-6
a_4	-1.286e-7
a_5	-1.17e-8

Table 3.9 Inverse Polynomial Coefficients of Sensor Nonlinearity

Coefficients	Values
c_0	6.999e-8
c_1	1.0
c_2	-1.642e-5
c_3	-1.368e-6
c_4	1.286e-7
c_5	1.17e-8

To illustrate the effect of VRE, this example assumes the system to be stationary with some undesirable vibration at frequency above the band of interest. The input $x(t)$ is defined as a sinusoidal signal with frequency w and amplitude$12.5m/s^2$.

$$x(t) = 12.5\sin(wt) \tag{3.34}$$

This input is used in $y(x)$ to obtain a distorted output $y(t)$.

$$y(t) = 1.054x10^{-4} + 12.4998\sin(wt) + 0.0002870\cos(2wt) \tag{3.35}$$
$$+0.0004485\sin(3wt) - 0.000392456\cos(4wt) - 0.00022316\sin(5wt)$$

When the inverse polynomial is applied to this function by applying $\hat{x}(y)$ the reconstructed input is given by.

$$\hat{x}(t) = -2.75x10^{-8} + 12.5\sin(wt) + 7.67\times10^{-8}\cos(2wt)$$
$$+2.63995\times10^{-8}\sin(3wt) - 1.137\times10^{-7}\cos(4wt) - 6.96212\times10^{-8}\sin(5wt) + \cdots \tag{3.36}$$

The key points of interest in this corrected signal is that the VRE has been significantly reduced, the vibration input has been restored to its original magnitude and other higher harmonics are reduced. Figure 3.23 shows the frequency spectrum of the corrected signal.

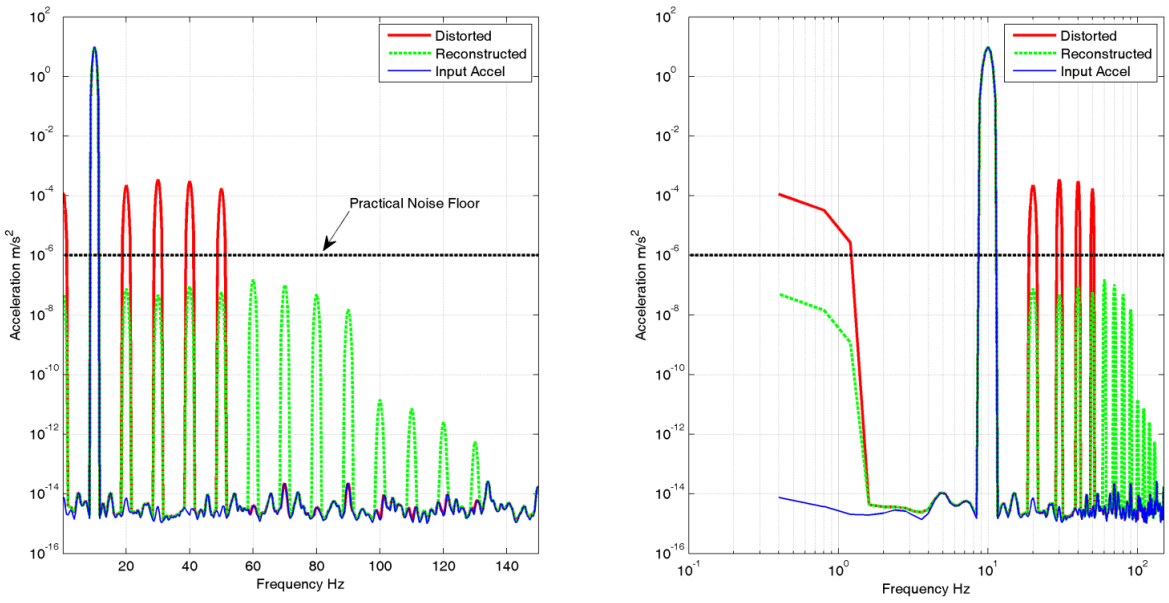

Figure 3.23 Polynomial Compensation of Full Bandwidth Data

Higher order harmonics (greater than input) occur as a result of the inaccuracy of the inverse polynomial. However in real systems, if the polynomial is accurate enough these harmonics will be reduced below the noise floor.

Even in the ideal compensation with a known system nonlinearity a perfect inverse cannot be generated within a finite order compensating polynomial. However the significant reduction in VRE and in the magnitude of the harmonics now permits the use of filters to allow for down-sampling.

3.3.3.2 Worst Case: Filtering of all vibration prior to compensation

First assume that no mechanical filter exists and all system accelerations are measured by the sensor. Its output will then have both harmonic and constant errors due to the nonlinearity according to equation (3.25). A filter with cutoff frequency f_0 which is below the vibration frequency w is used to remove the undesired vibration from y to get y' as shown in Figure 3.24.

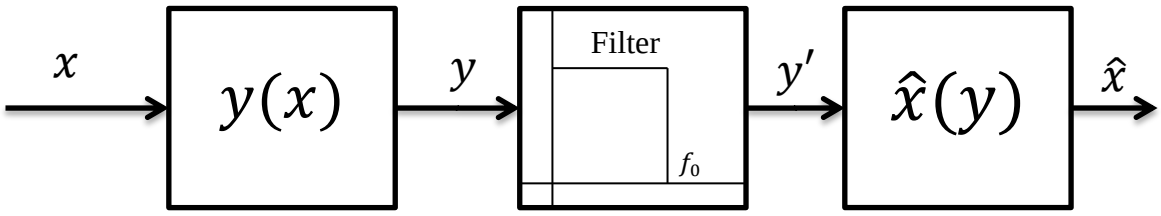

Figure 3.24 Compensation Block Diagram Filtered

In this case y' will take the following form.

$$y'(t) = 1.054 \times 10^{-4} \quad (3.37)$$

The VRE error which is represented by a DC term in equation (3.35) is still present in the signal after filtering. **This error can no longer be corrected because all information about the vibration has been eliminated.** This is the worst case, it will result in no correction of VRE.

3.3.3.3 Filtering of harmonics prior to compensation

This problem can also be represented by another situation. Similar to the above case, no mechanical filter is present, but in this case the bandwidth and sample rate are chosen to encompass the fundamental frequency of vibration but not the harmonics, i.e. $w < f_0 < 2w$. If the signal is filtered to remove the 2nd and higher order terms in equation (3.35) the result is.

$$y'(t) = 1.054 \times 10^{-4} + 12.4998 \sin(wt) \quad (3.38)$$

Now if the inverse polynomial is applied to this function the following reconstruction, plotted in Figure 3.25, is obtained.

$$\hat{x}(t) = 1.203 \times 10^{-7} + 12.5 \sin(wt) - 2.87 \times 10^{-4} \cos(2wt) \quad (3.39)$$
$$-4.48 \times 10^{-4} \sin(3wt) + 3.93 \times 10^{-4} \cos(4wt) + 2.23 \times 10^{-4} \sin(5wt)$$

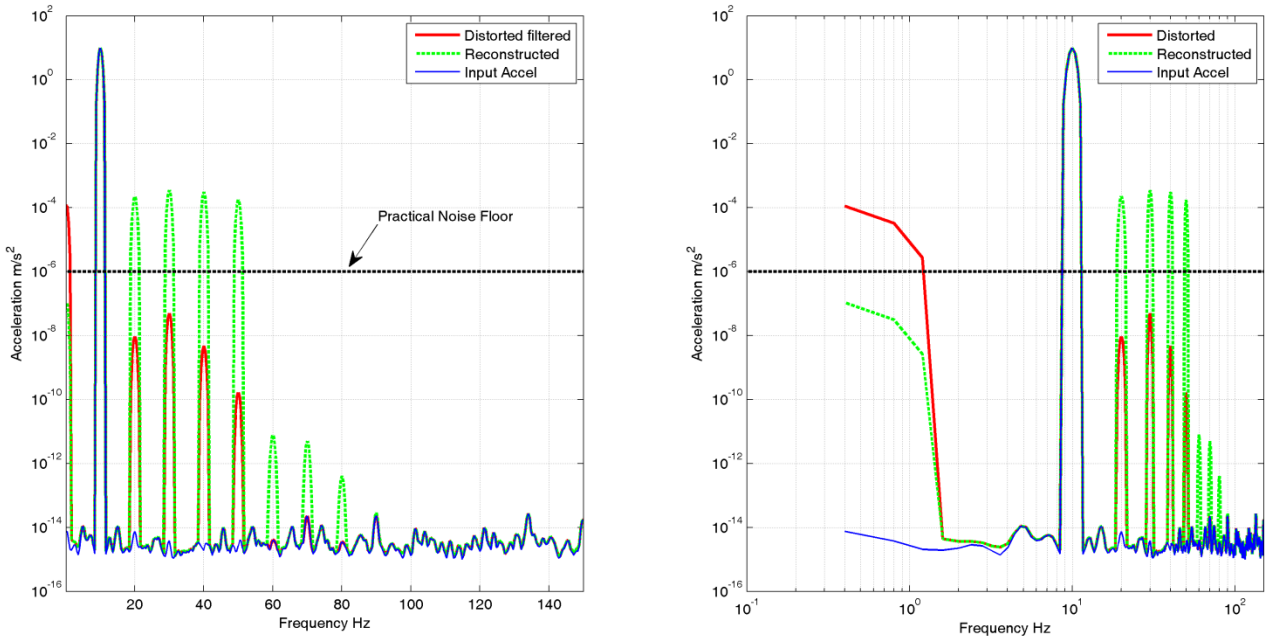

Figure 3.25 Polynomial Compensation of Reduced Bandwidth Data

While a reasonable reduction in VRE is achieved, the error is still 10 times that of the fully corrected solution. As expected the harmonics, or rather their inverses, which were filtered out previously reappear after correction. This effectively causes an increase of the bandwidth of the signal. If the sample rate is insufficient this bandwidth increase could result in aliasing or an increase if VRE due to insufficient sampling frequency.

3.3.4 Summary

Given the cases described above the following recommendations are given:

- Sample as fast as possible and maintain a broad bandwidth until after the nonlinearity correction.
- Reduce the system bandwidth before the accelerometer using a mechanical filter unless an accurate measurement of the vibration is required.

3.4 Implementation Issues with Mechanization

The previous sections have discussed the factors related to the conversion and compensation of the accelerometer output to obtain specific force. A similar process facilitates correcting gyros to measure angular rate. Once these signals are obtained it is necessary to integrate them in the inertial mechanization to obtain position, velocity, and attitude.

Prior to the design or selection of a computer or microprocessor, it is necessary to evaluate the algorithm to be used in the mechanization to determine the computational requirements. This is particularly important when designing a system for high vibration or dynamics environments as a high rate of computation is required to maintain the accuracy [2] [4].

The following equations and analysis represent a subset of that presented in [4]. They are used to demonstrate the errors encountered when high dynamics or vibrations are encountered. With such an analysis it should be possible to determine the minimum computation rate and the required accuracy of approximations.

3.4.1 K-Cycle Attitude Update

As shown in section 3.1.2, the updated direction cosine matrix can be calculated by integration of the differential equation (3.10). While it is possible to integrate the equation as written, it is a computational burden as 9 differential equations must be solved. Two main techniques exists to solve this problem. One uses quaternions to represent the DCM[1]

[1] Direction Cosine Matrix

and perform the computation. The other method, which will be discussed below calculates a rotation vector and corresponding rotation matrix to update the DCM. The derivation follows closely from [4].

Firstly, equation (3.10) is written as an integral over a single cycle, in this case the k-cycle which is the medium rate computation cycle.

$$\boldsymbol{C}_{k+1} = \boldsymbol{C}_k \exp\left[\int_{t_k}^{t_{k+1}} \boldsymbol{\Omega}\, \mathrm{dt}\right] \qquad \textbf{(3.40)}$$

Assuming the orientation of the angular rate vector ω remains nearly constant over the interval, the above equation can be written as:

$$\boldsymbol{C}_{k+1} = \boldsymbol{C}_k \exp[\boldsymbol{\sigma} \times] = \boldsymbol{C}_k \boldsymbol{A}_k \qquad \textbf{(3.41)}$$

$$\boldsymbol{\sigma} \times = \begin{bmatrix} 0 & -\sigma_z & \sigma_y \\ \sigma_z & 0 & -\sigma_x \\ -\sigma_y & \sigma_x & 0 \end{bmatrix} \qquad \textbf{(3.42)}$$

Where $\boldsymbol{\sigma} \times$ is a rotation vector[1] and $\boldsymbol{A}_k$ is a direction cosine matrix which transforms the DCM from time k to k+1. Expanding the exponential term in the above equation gives:

$$\boldsymbol{A}_k = \boldsymbol{I} + [\boldsymbol{\sigma} \times] + \frac{[\boldsymbol{\sigma} \times]^2}{2!} + \frac{[\boldsymbol{\sigma} \times]^3}{3!} + \frac{[\boldsymbol{\sigma} \times]^4}{4!} \qquad \textbf{(3.43)}$$

The exact representation of this equation can also be written as follows:

$$\boldsymbol{A}_k = \boldsymbol{I} + \frac{\sin \sigma}{\sigma}[\boldsymbol{\sigma} \times] + \frac{1 - \cos \sigma}{\sigma^2}[\boldsymbol{\sigma} \times]^2 \qquad \textbf{(3.44)}$$

[1] It is called a rotation vector, but in skew symmetric form it is a matrix

$$\sigma = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \quad \textbf{(3.45)}$$

Equation (3.44), when used in equation (3.41) allows for the update of the direction cosine matrix with significantly lower computational time. The requirement is that the angular rates must be integrated to form a rotation vector. This integration is performed in the j cycle.

When implementing this rotation update algorithm in a computer, especially one with limited capability, it is extremely important to avoid the operations which are time consuming. Among the operations which are excessively long are division and trigonometric functions. For this reason approximations are used to reduce computation time.

$$a_1 = \frac{\sin\sigma}{\sigma} = 1 - \frac{\sigma^2}{3!} + \frac{\sigma^4}{5!} \ldots \quad \textbf{(3.46)}$$

$$a_2 = \frac{1 - \cos\sigma}{\sigma^2} = \frac{1}{2!} - \frac{\sigma^2}{4!} + \frac{\sigma^4}{6!} \ldots \quad \textbf{(3.47)}$$

Further simplifications can be made by truncating the series expansion. While this truncation can significantly reduce computation time, the accuracy of the rotation update can be significantly reduced under some situations. Based on equation (3.44), the DC angular drift error can be calculated for a single axis of rotation [4].

$$D_{dc} = \frac{1}{\delta t}(\sigma\, a_1 \cos(\sigma) - \sin(\sigma) + \sigma^2 a_2 \sin(\sigma) \quad \textbf{(3.48)}$$

where:

D_{dc} is the angular drift after one cycle

σ is the angular increment

δt is the update interval

$a_1 = 1, a_2 = 0$ first order algorithm

$a_1 = 1, a_2 = 0.5$ second order algorithm

$a_1 = 1 - \frac{\sigma^2}{6}, a_2 = 0.5$ third order algorithm

$a_1 = 1 - \frac{\sigma^2}{6}, a_2 = 0.5 - \frac{\sigma^2}{24}$ fourth order algorithm

The following table shows the attitude drift error for a selection of angular rates, sample rates and algorithm orders. This table represents the error in the k-cycle update for continuous attitude changes, primarily due to dynamics. It may not be affected by vibration if it is zero mean, and at a frequency much higher than the update rate. This does however require a rotation vector which has been computed in the j cycle at a rate higher than the vibration. So the conclusion is that the k-cycle update speed is linked to the rate of dynamics, where the j cycle must account for vibration.

Table 3.10 K-Cycle Attitude Drift Errors

	Attitude Drift Error $°/hour$			
Order	$\omega = 0.1\ rad/s$ $\delta t = 0.01s$	$\omega = 0.1\ rad/s$ $\delta t = 0.001s$	$\omega = 10\ rad/s$ $\delta t = 0.01s$	$\omega = 10\ rad/s$ $\delta t = 0.001s$
1	0.006875	6.875e-5	6868	68.75
2	0.003437	3.43e-5	3427	34.37
3	6.83e-10	2.55e-12	6.86	6.87e-4
4	1.7e-10	2.64e-12	1.7	1.7e-4

3.4.2 J-Cycle Rotation Angle Computation

The previous section presented an equation for computing the direction cosine matrix update based on the rotation vector $\boldsymbol{\sigma}$ and discussed the accuracy of that calculation. In this section I will discuss the calculation of the rotation vector in the high speed processing part of the mechanization. The rotation vector is effectively the net change in attitude from

time t_k to t_{k+1}. Computation of the rotation vector is effectively integrating the angular rate measurements. The high speed j-cycle is used for the computation of the rotation vector angle. If the direction of the angular rate $\boldsymbol{\omega}$ remains unchanged over the interval, the rotation angle $\boldsymbol{\sigma}$ is determined simply as the integration of $\boldsymbol{\omega}$ over the k cycle time.

$$\boldsymbol{\sigma} = \int_{t_k}^{t_{k+1}} \boldsymbol{\omega}\, dt \qquad \textbf{(3.49)}$$

If however the rotation direction does not remain constant the issue of non-commutativity[1] of the rotations becomes a problem. To address this issue the following algorithm is proposed [5].

$$\boldsymbol{\sigma} = \boldsymbol{\alpha}_{k+1} + \boldsymbol{\delta\alpha}_{k+1} \qquad \textbf{(3.50)}$$

where

$$\boldsymbol{\alpha} = \int_{t_k}^{t_{k+1}} \boldsymbol{\omega}\, dt$$

$$\boldsymbol{\delta\alpha}_{k+1} = \int_{t_k}^{t_{k+1}} \boldsymbol{\alpha} \times \boldsymbol{\omega}\, dt$$

The above method essentially truncates the equation for the rotation vector to produce a simple algorithm suitable for high rate integration. As this is a numerical integration, higher processing rates reduce the errors due to approximation. The j cycle is also used for the integration of acceleration to obtain a velocity increment.

[1] The order of rotations is important

3.4.3 I-Cycle Rotation Updates

As stated in [4], the i-cycle is a slow cycle used for the update of the navigation frame which changes at a very slow rate. The equations are essentially the same as the k cycle mentioned above. The following equation shows the continuous form of the navigation frame update. This is used to update the DCM relating the navigation to the earth frame. This change would be a result of the change of position.

$$\dot{\boldsymbol{C}}_n^e = \boldsymbol{C}_n^e \boldsymbol{\Omega}_{en}^n \qquad \textbf{(3.51)}$$

where

$$\boldsymbol{\Omega}_{en}^n = \boldsymbol{\omega}_{en} \times \qquad \textbf{(3.52)}$$

$$\boldsymbol{\omega}_{en} = \left[\frac{v_e}{R_0 + h} \quad \frac{-v_n}{R_0 + h} \quad \frac{-v_e \tan(L)}{R_0 + h}\right]^T \qquad \textbf{(3.53)}$$

v_e = East Velocity

v_n = North Velocity

L = Latitude

R_0 = Radius of Earth

h = Height above earth surface

The rotation of this matrix is done in an identical way to that for the body to navigation frame matrix using equation (3.44).

3.5 Velocity and Position Updates

Once the attitude is computed, it is possible to transform the specific force measurements into the navigation frame and integrate to get the new velocity and position.

$$\boldsymbol{f}^n = \boldsymbol{C}_b^n \boldsymbol{f}^b \qquad \textbf{(3.54)}$$

The new velocity can be expressed using the following equation.

$$\boldsymbol{v}_{k+1}^{n} = \boldsymbol{v}_{k}^{n} + \int_{t_k}^{t_{k+1}} \boldsymbol{f}^{n}\, dt - \int_{t_k}^{t_{k+1}} [2\boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}] \times \boldsymbol{v}_{k}^{n}\, dt + \int_{t_k}^{t_{k+1}} \boldsymbol{g}^{n}\, dt \qquad \textbf{(3.55)}$$

where:

$\boldsymbol{f}^{n}$ specific force acceleration in navigation frame

$\boldsymbol{\omega}_{ie}$ Earth rotation vector: Rotation of Earth frame with respect to inertial frame

$\boldsymbol{\omega}_{en}$ Transport rate: Rotation of Navigation frame with respect to earth frame

$\boldsymbol{v}^{n}$ velocity in navigation frame

$\boldsymbol{g}^{n}$ gravity vector in navigation frame

Position at this point can be integrated to find the change in position in the navigation frame. In general this is used to compute a change in latitude, longitude and altitude.

$$\boldsymbol{\delta x}_{k+1}^{n} = \int_{t_k}^{t_{k+1}} \boldsymbol{v}^{n}\, dt \qquad \textbf{(3.56)}$$

Although not shown, the integration of specific force requires special consideration to account for rotation of the body during the k-cycle. This process is examined in detail in [4].

3.6 Conclusions

This chapter has demonstrated a number of system errors. The focus on vibration, noise and delay related errors has led to some recommendations for compensation and requirements for system design.

To combat vibration rectification from nonlinearity and from computational errors, a high sample rate is required. This sample rate must, for large vibrations, be several times the high end of the frequency bandwidth. It is at this high frequency that the compensations for nonlinearity and integration must take place.

Noise must be reduced whenever possible, however considerations must be made for different types of noise which impact the system in different ways. For example some amplifiers with lower noise specs may use more power and cause increased thermal instability and drift.

Delay errors result from many sources such as poorly synchronized ADCs, mismatched filter components, and poor timing resolution. This error can cause problems for many mechanization algorithms and especially for calibration in dynamic testing. Care must be taken to identify, characterize, and if possible minimize these synchronization delays.

Finally, an overview of the inertial mechanization has demonstrated the need for high speed processing when fast dynamics are present. The presence of integration and rotation approximations introduces errors which are magnified when signals are not stationary.

The overall conclusion from this chapter is that high sample rates and high processing rates are required to measure high dynamics or vibration. If the dynamics of the base signal are slow but have higher frequency vibration, it is only necessary to run the ADC and nonlinearity compensators at high sample rate. It is then possible to filter the data to allow for lower processing rates.

CHAPTER 4: IDENTIFICATION OF NONLINEARITY AND ESTIMATION OF PARAMETERS

The previous discussion of sensor errors has focused on their nonlinearity. The result of nonlinearity is a DC error related directly to the acceleration, or in the case of a gyro, angular rate.

To mitigate these errors, it is necessary to obtain a model for the sensor to correct the output. As previously mentioned, a commonly used nonlinearity model is the polynomial. Its simplicity and adaptability make it useful in many applications. The standard means by which this model is identified are multipoint and centrifuge testing. Multipoint tests use a "dividing head" to apply fractions of gravity to the accelerometer to obtain a polynomial in the range $\pm 1g$. The centrifuge test uses rotation to apply acceleration from zero to the limit of the accelerometers range [1]. The main disadvantage of the multipoint test is the limited range, although the accuracy is very good. The main disadvantage of the centrifuge is the necessity to stop and turn the accelerometer around for negative acceleration.

Another less conspicuous problem is the one of self-heating. As discussed in [20], centrifuge and multipoint testing applies a DC acceleration to the sensor. For force feedback servo accelerometers the rebalance force is proportional to the acceleration. In these devices self-heating of the torquer and electronics can cause bias shifts. The result of these shifts will effectively add to the nonlinearity during static acceleration tests like centrifuge calibration. Further, at different acceleration levels, different settling temperatures will be experienced. This will result in the improper estimate of the entire nonlinearity for a given temperature. A representative block diagram of an accelerometer, with self-heating taken into account, is shown in Figure 4.1.

The method proposed in this thesis seeks to calibrate the a portion of the range of the accelerometer in the positive and negative directions while minimizing the effects of self-heating. By applying a short duration vibration to the sensor using a linear exciter, this range of the accelerometer can be tested in one burst. The short duration of the burst prevents any significant heat generation. This calibration can then be performed at different steady temperatures to determine the variation of bias and nonlinear terms with temperature. Similarly, different frequencies can be tested to determine the effect of system dynamics on nonlinearity.

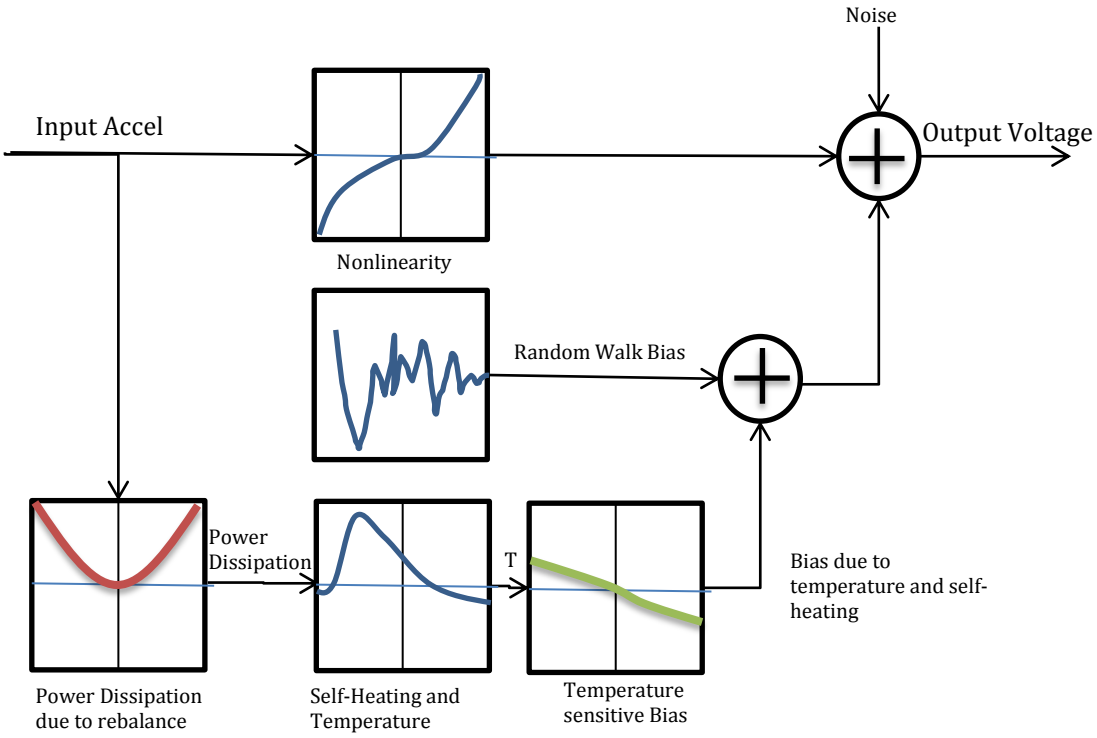

Figure 4.1 Accelerometer Model with Self-heating

The process of identification and parameter estimation with the goal of producing a compensator is not a straight path. It requires some iteration and alternate paths. The following flow chart provides a road map to this process.

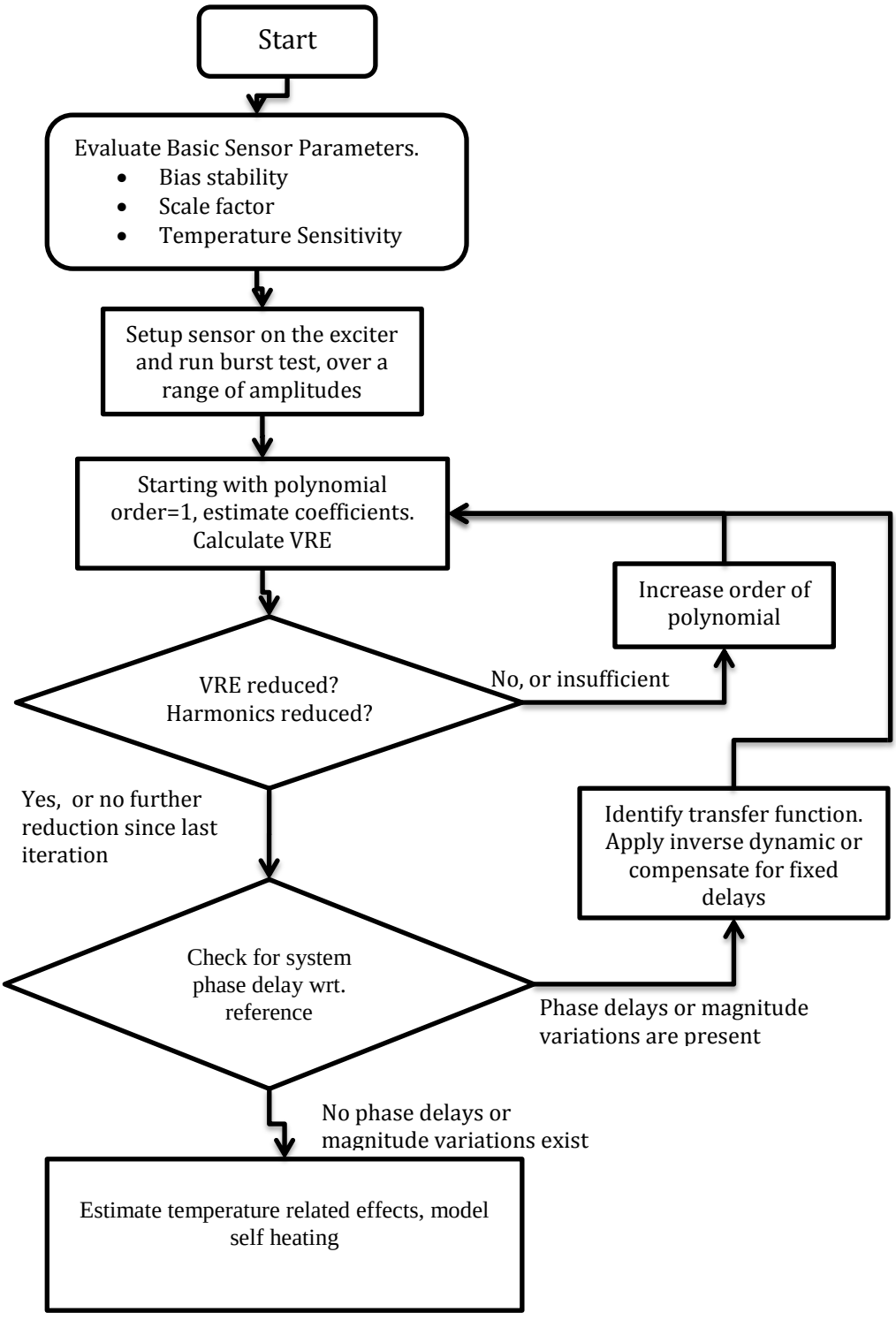

Figure 4.2 Sensor Identification Flow Chart

4.1 Experimental Results

The following experimental results demonstrate the effectiveness of the linear exciter as a calibration tool in the investigation on nonlinearity. The equipment used was an Aerotech air bearing linear stage, a Polytec laser vibrometer, and a custom built FPGA based data acquisition system developed in this research, capable of continuous high sample rate recording. The results of three of the sensors are shown for the tested range $\pm 1g$. All are examples of very short duration testing with 3 short bursts at 10 Hz. The data in this case was analyzed using least squares. The formulation is described later along with a review of the least squares and Kalman filtering methods.

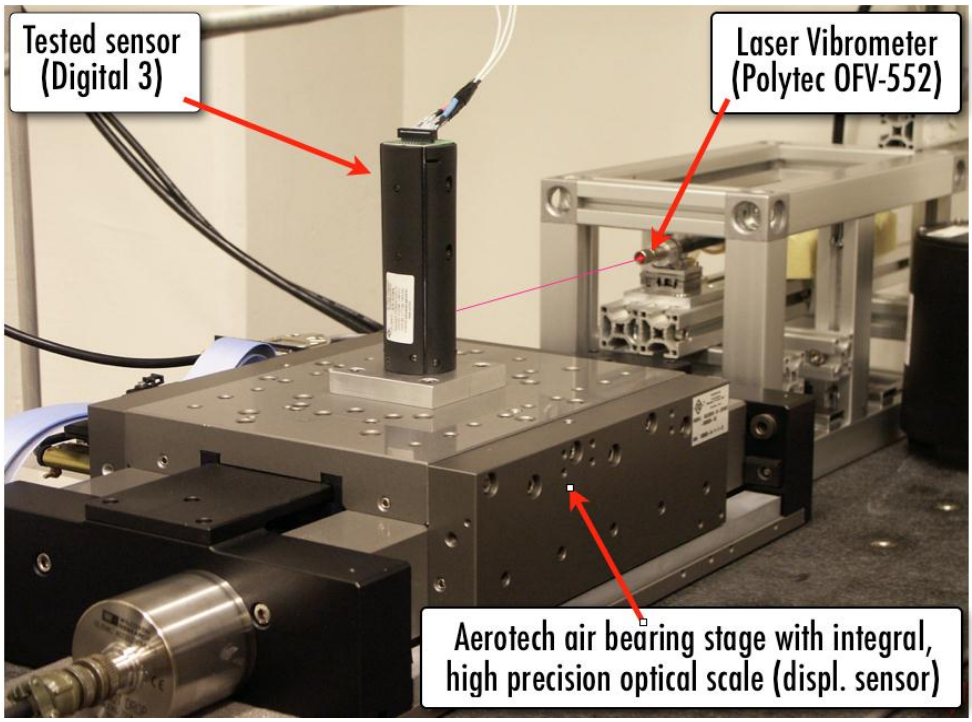

Figure 4.3 Burst Test Setup

4.1.1 Investigation of PCB and Kistler MEMS accelerometers

The first set of tests shown here are burst tests of the Kistler and PCB accelerometers. These accelerometers have specifications of 0.2 % and 1 % FSR nonlinearity respectively. Both are rated for $\pm 3g$.

A 10 Hz sinusoidal burst of 1g amplitude is applied to both accelerometers. Three different burst periods are used to verify that the velocity shift is proportional to the period of vibration. The output is integrated resulting in the velocity profile shown in Figure 4.4.

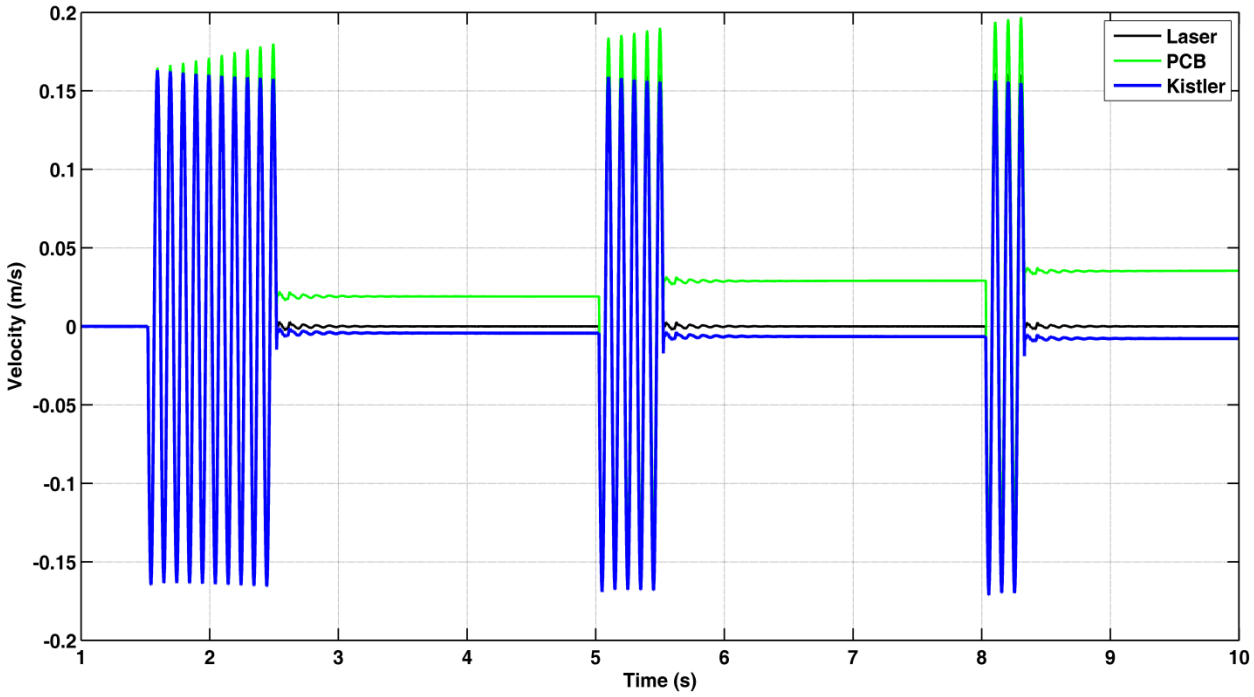

Figure 4.4 Burst Tests of PCB and Kistler with Uncorrected Velocity

Least squares estimation was used to determine the nonlinear polynomial, the coefficients of which are shown in Table 4.1. From this polynomial the vibration rectification vs amplitude graph was plotted as seen in Figure 4.5. Of interest in this plot is the crossover of the Kistler Servo accelerometer. It appears that at certain amplitudes no vibration rectification will occur.

Table 4.1 Non-Linear Coefficients for PCB and Kistler

Coefficients	Kistler	PCB
a_0	0.01262	0.002145
a_1	1.02559	0.9725
a_2	-0.00101	-8.68e-5
a_3	-6.558e-4	2.457e-4
a_4	9.4611e-6	5.4568e-6
a_5	6.175e-6	4.4478e-7

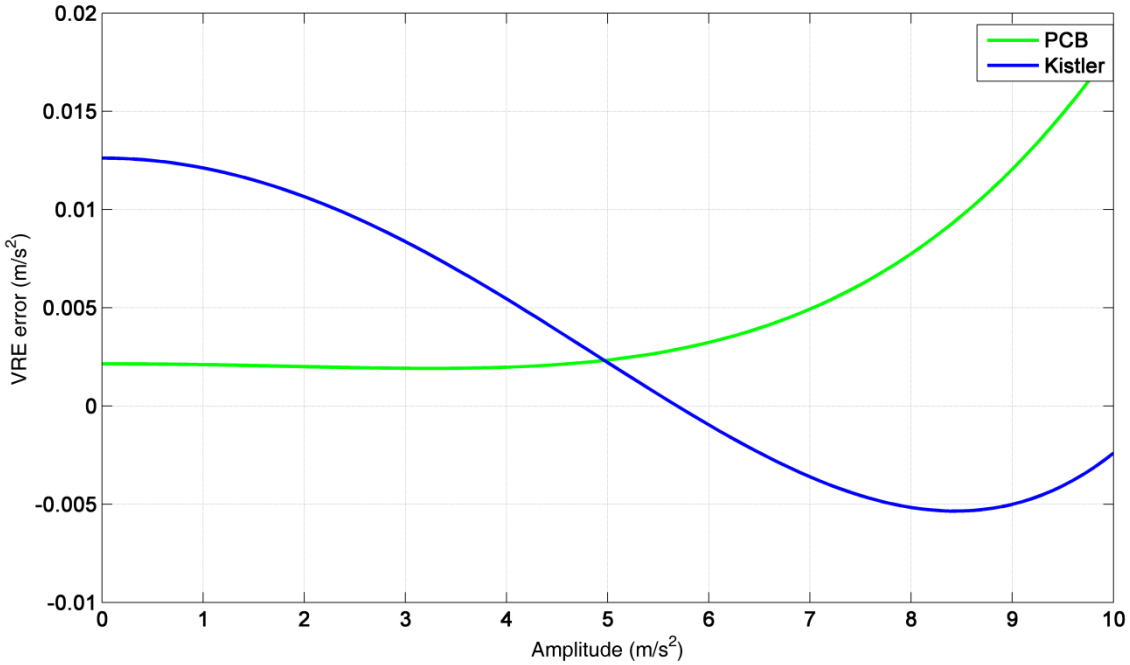

Figure 4.5 VRE vs Amplitude PCB and Kistler

The estimated nonlinearity is used to correct the accelerometer output and again the result is integrated to velocity.

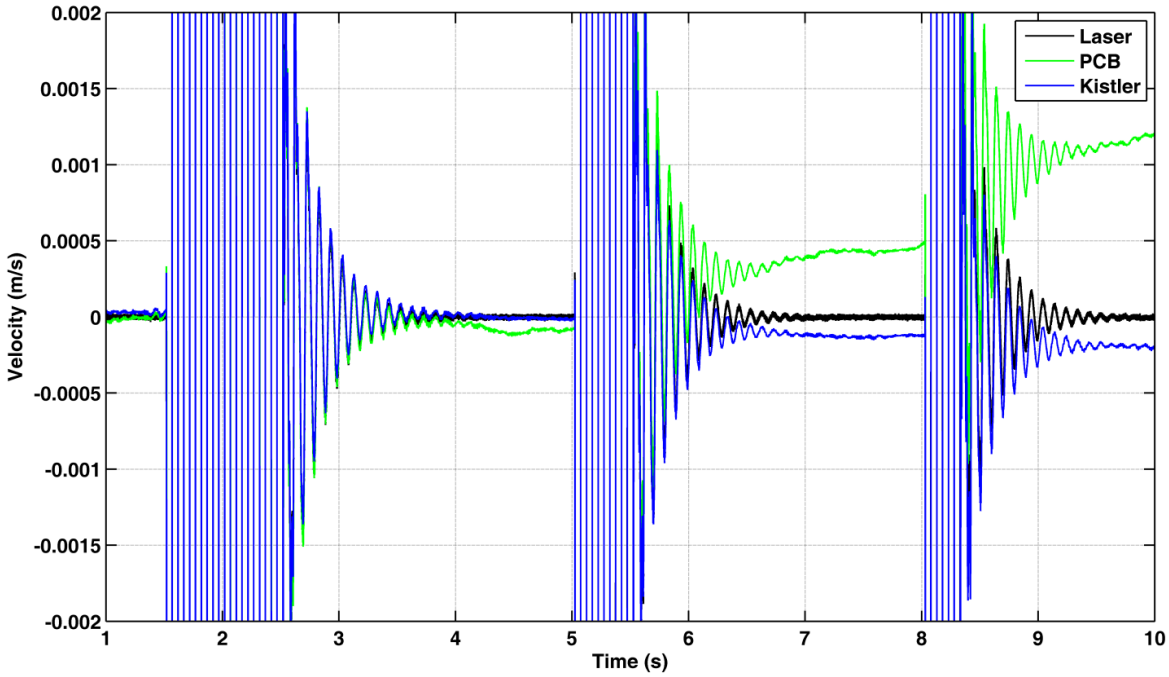

Figure 4.6 Burst Test of PCB and Kistler with Corrected Velocity

Figure 4.6 shows the result after correcting the nonlinearity. The vibration rectification error is reduced 100 to 1000 times leaving only bias drift errors. Something to note is the drift of the PCB whose slope changes after each burst. This may be a result of self-heating even though it is not a force rebalanced accelerometer. The heat may be a result of power dissipation in the amplifiers which output the signal.

4.1.2 Investigation of QA700

A similar test was performed for the QA700. This time, multiple amplitudes of excitation were tested. Vibration rectification from this accelerometer is significantly lower than either of the MEMs sensors test above even though the sensors are similar in cost. The figure below demonstrated the combined effect of drift and vibration rectification. The blue line is the estimated drift from the least squares estimation. Figure 4.8 shows the same data with the estimated drift subtracted to clearly show the predictable vibration

rectification. The staircase step size was used to directly measure the vibration rectification.

The "corrected integrated accel" indicates the signal if drift and VRE are removed. The degree of matching between it and the laser indicates the accuracy of the model for estimation. Thus it can be claimed that a fixed nonlinearity and a bias drift are the major components of the error and no significant change in the nonlinearity occurs over the length of the test.

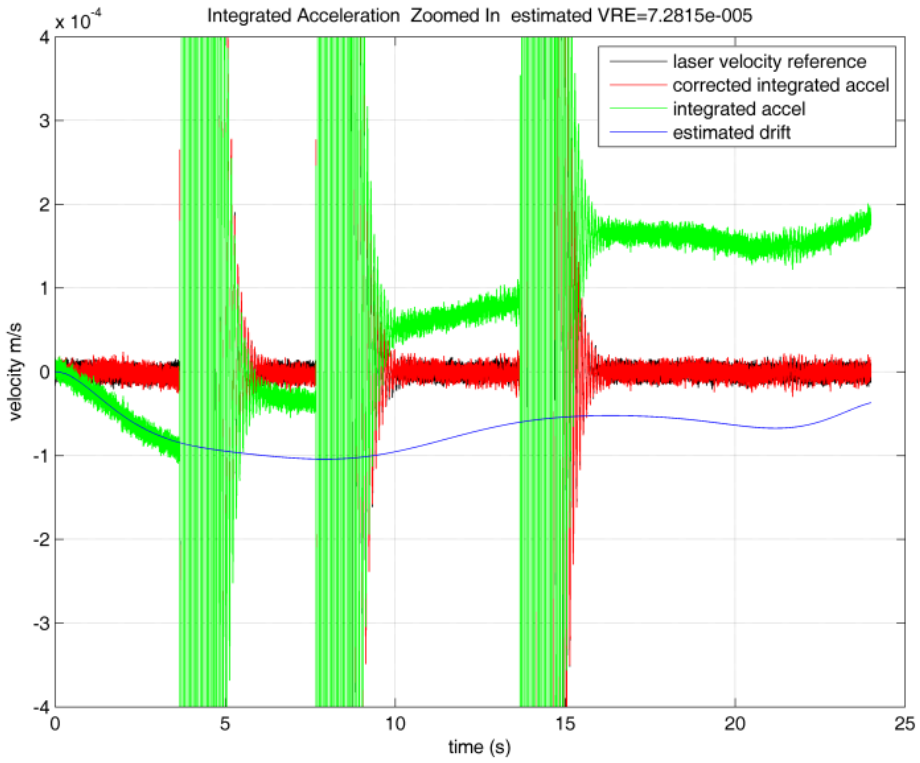

Figure 4.7 Burst Test of QA700 With Drift

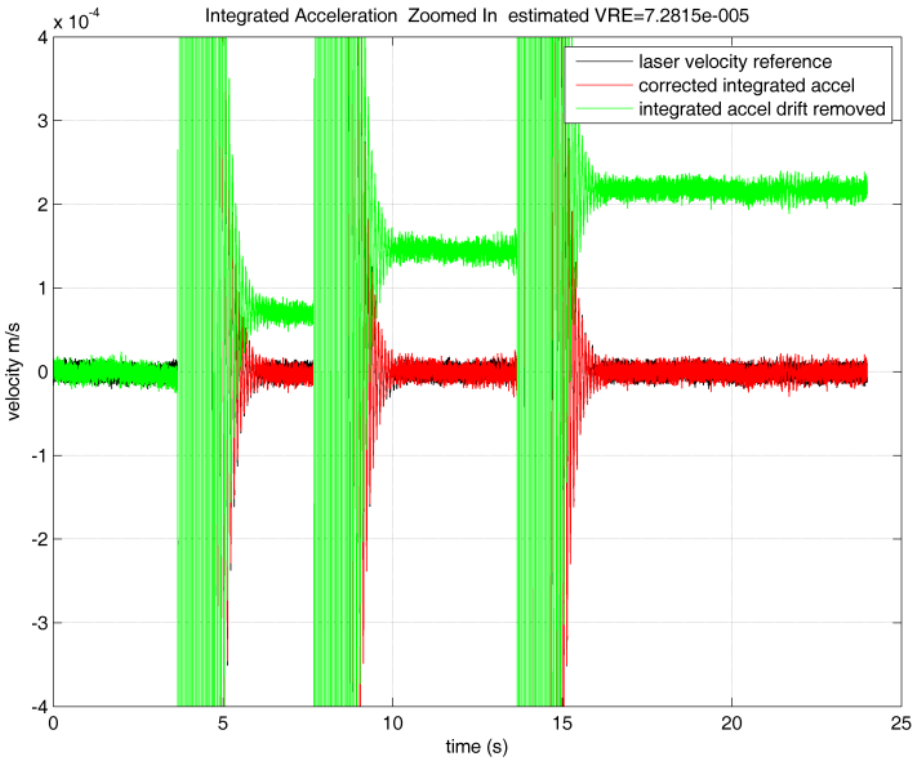

Figure 4.8 Burst Test QA700 Drift Removed

Following all the tests a VRE vs Amplitude graph, Figure 4.9, was generated showing the comparison between the measured VRE and the analytically predicted VRE. It can be used to determine the accuracy of the estimated polynomial. It is of particular interest that each amplitude of excitation produced a different polynomial. Although following the measured trend reasonably, each is valid only for a subset of the range. The highest amplitude test obviously covers the widest range. The lowest amplitude test does not well reflect the experimental data. This may be a result of noise or other dynamics which are normally overshadowed by large excitations. Using the polynomial from the largest magnitude test does seem to reliably reduce the VRE to well below the bias drift level. Future testing will be necessary to check for variations of the curves at different temperatures and frequencies.

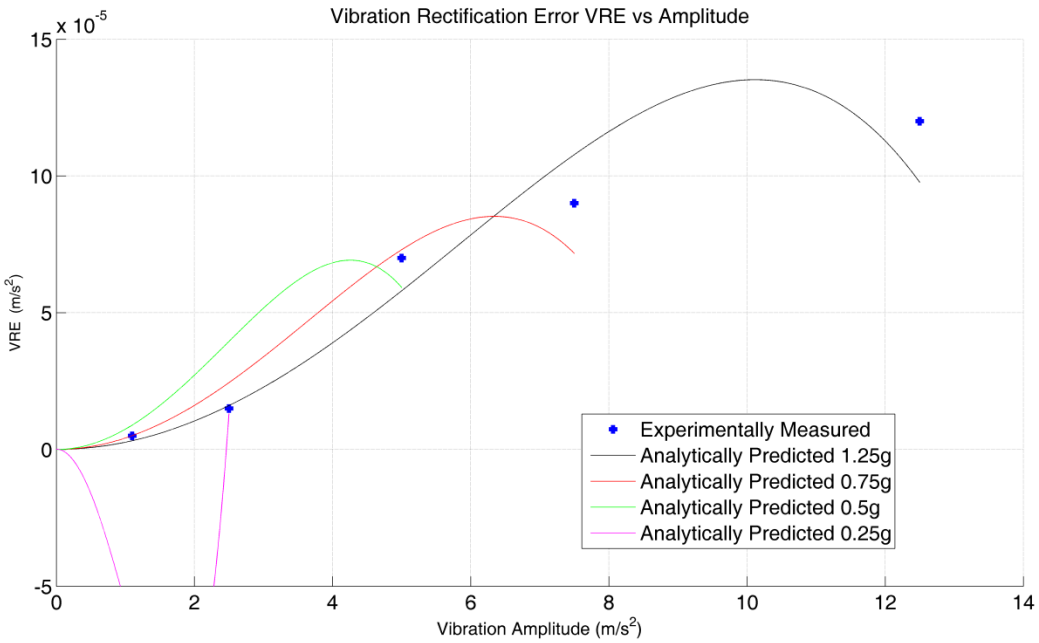

Figure 4.9 VRE vs Amplitude for QA700

The most representative calibration from above was used to correct other data sets such as the one shown in Figure 4.10 and Figure 4.11. This test uses just two 1 second bursts at 1g pk-pk. Here the displacement error is shown before compensation in Figure 4.10, and after in Figure 4.11. Although the total deviation after compensation is around 0.1 mm, the vibration related drift accounts for only 0.01 mm indicating the success of the compensation.

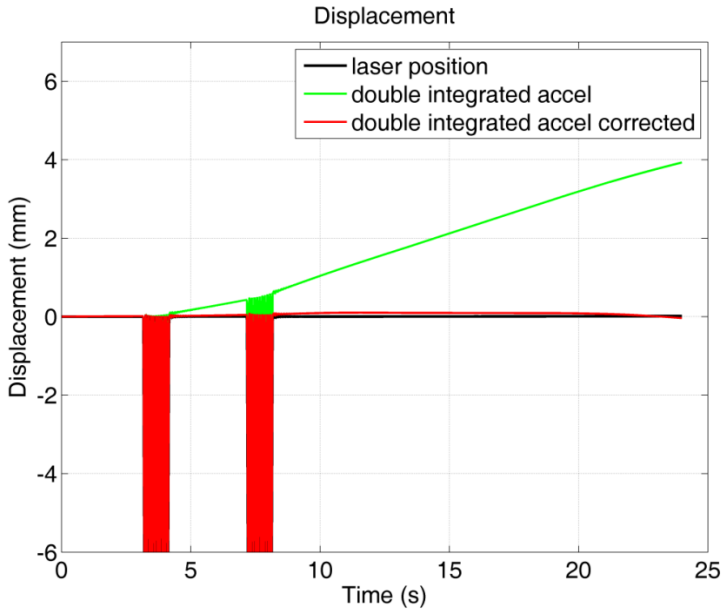

Figure 4.10 QA700 Burst Test Displacement Error

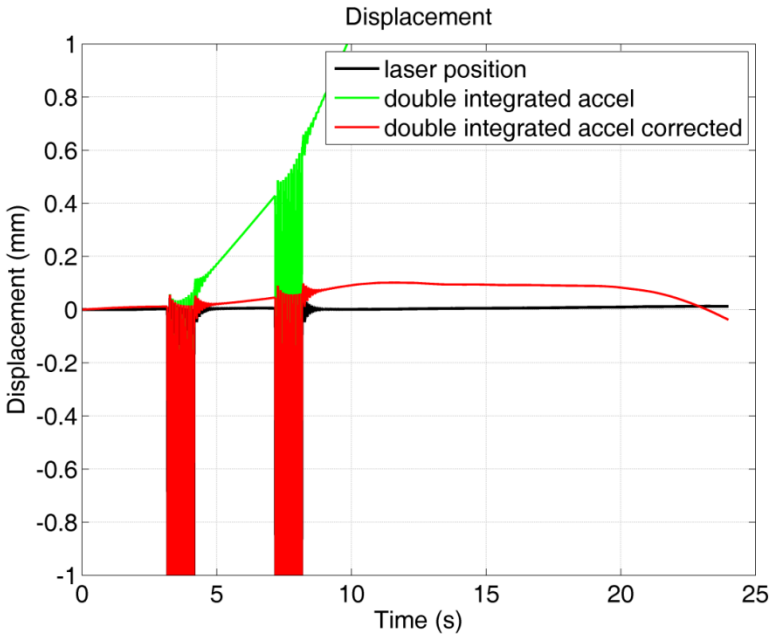

Figure 4.11 QA700 Burst Test Displacement Error Zoomed

These test results have indicated that this type of calibration can estimate the nonlinearity well enough to reduce VRE to well below that of the bias drift. In the above example the corrected signal shows no noticeable effect from VRE, however bias drift is still clearly an

issue. It may be possible to attribute this, at least partially, to temperature sensitivity and thus reduce it further by calibration. Beyond that, having reduced VRE, it is now possible to track the bias drift with a Kalman filter as is normally done in aided inertial systems.

4.2 Estimation Review

4.2.1 Least Squares

The method of least squares is used to estimate the unknown coefficients in mathematical models given noisy data [27]. Least squares has the advantage of finding the optimal solution for over-determined systems of equations. The basis for the linear least squares problems must take the following form

$$z = \boldsymbol{H}x \tag{4.1}$$

where $\boldsymbol{H}$ is the design matrix, x is a vector of unknowns, and z are the measurements. In order to solve for x, an equation is written based on the previous one where a value $\hat{x}$ being an estimate of x is used to minimize the sum of squared measurement errors ε.

$$\varepsilon^2 = |\boldsymbol{H}\hat{x} - z|^2 \tag{4.2}$$

The minimum error is found where its derivatives with respect to $\hat{x}$ are zero.

$$\frac{\partial \varepsilon}{\partial \hat{x}} = 0 = 2\boldsymbol{H}^T[\boldsymbol{H}\hat{x} - z] = 2\boldsymbol{H}^T\boldsymbol{H}\hat{x} - 2\boldsymbol{H}^T z \tag{4.3}$$

$$\hat{x} = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T z \tag{4.4}$$

This form of the equation is called the *normal equation*. The assumption of this equation is that all measurements are equally weighted. A solution exists when the Gramian matrix

$$\boldsymbol{\mathcal{G}} = \boldsymbol{H}^T\boldsymbol{H} \tag{4.5}$$

is nonsingular. The least squares solution can be extended by weighting each measurement. This approach is described in [28]. The form of this equation is given as follows

$$\hat{x} = (\boldsymbol{H}^T \boldsymbol{R}^{-1}\, \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{R}^{-1}\, z \qquad (4.6)$$

where **R** is the weighting matrix. The computation of $\boldsymbol{R}^{-1}$ becomes increasingly time consuming as the number of measurements increases. This problem can be resolved using recursive least squares which sequentially updates the solution with new measurements rather than doing so all at once. For consistency, the equations below are written using the same notations as in Kalman filtering.

$$\boldsymbol{K} = \boldsymbol{P}_{\mathrm{k}} \boldsymbol{H}_{\mathrm{k+1}}^T \left(\boldsymbol{R}_{\mathrm{k+1}} + \boldsymbol{H}_{\mathrm{k+1}}\, \boldsymbol{P}_{\mathrm{k}} \boldsymbol{H}_{\mathrm{k+1}}^T\right)^{-1} \qquad (4.7)$$

$$\boldsymbol{P}_k = (\boldsymbol{H}_k^T \boldsymbol{R}_k^{-1} \boldsymbol{H}_k)^{-1} \qquad (4.8)$$

$$\boldsymbol{P}_{\boldsymbol{k+1}} = (\boldsymbol{I} - \boldsymbol{K}\, \boldsymbol{H}_{\mathrm{k+1}}\ )\boldsymbol{P}_{\mathrm{k}} \qquad (4.9)$$

$$\hat{x}_k = \hat{x}_k + \boldsymbol{K}_k (z_k - \boldsymbol{H}_k \hat{x}_k) \qquad (4.10)$$

The Kalman gain matrix $\boldsymbol{K}$ is calculated in the same way as in Kalman filtering.

4.2.2 Kalman Filter

Another common estimation method, which can be adopted for parameter identification, is Kalman filtering. Originally Kalman filters were developed as optimal observers for the states of linear dynamic systems. Its adaptability and functionality in control systems and navigation applications have led to its widespread use in nearly every GPS and inertial system currently in use. The following overview presents the basic principles and

equations as well as how Kalman filters can be used for the estimation of model parameters. A comprehensive discussion can be found in [27].

A common method of expressing the system model uses the following state space form.

$$\dot{\boldsymbol{x}} = \boldsymbol{F}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{G}\boldsymbol{w} \tag{4.11}$$

where $\boldsymbol{x}$ is the state vector, $\boldsymbol{F}$ is the dynamics matrix, $\boldsymbol{B}$ is the input coupling matrix, $\boldsymbol{u}$ is the input vector, $\boldsymbol{G}$ is the noise coupling matrix, and $\boldsymbol{w}$ is the system noise vector. This system is usually converted to the discrete form for use in the Kalman Filter and takes the following form.

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}_{k,k+1}\boldsymbol{x}_k + \boldsymbol{u}_k + \boldsymbol{w}_k \tag{4.12}$$

where $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k+1}$ are the current and future time steps of the state vector, $\boldsymbol{\Phi}_{k,k+1}$ is the state transition matrix which propagates the state vector from time k to k+1. The state transition matrix is the fundamental solution to the homogeneous differential equation.

$$\dot{x} = \boldsymbol{F}x \tag{4.13}$$

$$\boldsymbol{\Phi}(t,\tau) = e^{F(t-\tau)} \tag{4.14}$$

An approximation of this matrix exponential, based on Taylor series expansion, is commonly used.

$$e^{\boldsymbol{F}(t-\tau)} = \boldsymbol{I} + \boldsymbol{F}(t-\tau) + \frac{\boldsymbol{F}^2(t-\tau)^2}{2!} + \cdots \tag{4.15}$$

The Kalman filter uses a combination of prediction and correction to obtain an estimate of the state vector. The basic steps of the algorithm are as follows:

1. Extrapolate the a priori estimate of the state vector $\hat{x}_{k+1}^{-}$ by multiplying the state transition matrix Φ with the a posteriori estimate from the previous time step $\hat{x}_{k}^{+}$

$$\hat{x}_{k}^{-} = \boldsymbol{\Phi}\, \hat{x}_{k-1}^{+} \quad (4.16)$$

2. Extrapolate the apriori estimate of the error covariance $\boldsymbol{P}_{k+1}^{-}$

$$\boldsymbol{P}_{k}^{-} = \boldsymbol{\Phi}\, \boldsymbol{P}_{k-1}^{-}\, \boldsymbol{\Phi}^{T} + \boldsymbol{Q}_{k-1} \quad (3.17)$$

3. Compute the Kalman gain matrix $\boldsymbol{K}_k$

$$\boldsymbol{K}_k = \boldsymbol{P}_k^{-}\boldsymbol{H}_k^{T}(\boldsymbol{R}_k + \boldsymbol{H}_k\boldsymbol{P}_k\boldsymbol{H}_k^{T})^{-1} \quad (4.17)$$

4. Update the a posteriori estimate of the error covariance $\boldsymbol{P}_k^{+}$

$$\boldsymbol{P}_k^{+} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_k^{-} \quad (4.18)$$

5. Update the a posteriori estimate of the state vector $\hat{x}_k^{+}$

$$\hat{x}_k^{+} = \hat{x}_k^{-} + \boldsymbol{K}_k(z_k - \boldsymbol{H}_k\hat{x}_k^{-}) \quad (4.19)$$

Although it is possible to predict and update at the same rate as described by these equations, in practice, for navigation applications the measurement updates are infrequent compared to the prediction steps. It must also be noted that this implementation assumes a linear time invariant model. In many problems, the system is neither time invariant nor linear. This leads to a method of Kalman filtering referred to as the extended Kalman filtering where the assumption is made that the model is sufficiently linear over a small enough time frame. It is then linearized about the current estimate of the state vector. Examples of this can be found in [27]

4.3 Estimation Formulations

For both of the estimation examples the solution is formulated to compute parameters from the burst test.

4.3.1 Least Squares

In order to solve for the nonlinearity of the accelerometer using least squares, the problem is formulated in the following way. Defined in equation (3.16), $y(t)$ is the nonlinear output of the accelerometer. Assuming that a_{ref} is the acceleration reference, a nonlinear polynomial $F(y(t))$ is introduced which will correct the distorted output.

$$F\big(y(t)\big) = a_{ref} \qquad (4.20)$$

To account for the bias drift and offset of the accelerometer a component $B(t)$ is introduced, which in this research is assumed to be a polynomial. The order of this polynomial is dependent on the length of the data set, and the magnitude of the bias drift variations. When broken down into individual coefficients and adding the bias drift terms equation (4.20) becomes.

$$c_0 + c_1 y(t) + c_2 y(t)^2 + c_3 y(t)^3 \ldots + b_1 t + b_2 t^2 + b_3 t^3 \ldots = a_{ref} \qquad (4.21)$$

The equation is then written in a matrix form to match the form of equation (4.1). The time sequence is broken into discrete times of period Δt corresponding to the sample period.

$$\begin{bmatrix} 1 & y(t_0) & y(t_0)^2 & y(t_0)^3 & \ldots & t_0 & t_0^2 & t_0^3 & \ldots \\ 1 & y(t_1) & y(t_1)^2 & y(t_1)^3 & .. & t_1 & t_1^2 & t_1^3 & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & y(t_{n-1}) & y(t_{n-1})^2 & y(t_{n-1})^3 & .. & t_{n-1} & t_{n-1}^2 & t_{n-1}^3 & \ldots \\ 1 & y(t_n) & y(t_n)^2 & y(t_n)^3 & \ldots & t_n & t_n^2 & t_n^3 & \ldots \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ \ldots \\ b_1 \\ b_2 \\ b_3 \\ \ldots \end{bmatrix} = \begin{bmatrix} a_{ref}(t_0) \\ a_{ref}(t_{10}) \\ \ldots \\ a_{ref}(t_{n-1}) \\ a_{ref}(t_n) \end{bmatrix} \qquad (4.22)$$

By using the above formulation the vector of unknown coefficients for the nonlinear polynomial and the bias drift can be estimated. This version of the formulation is designed

to solve for the compensating polynomial but it can be written to solve for the "forward nonlinearity" as well.

It is also possible to write the least squares observation model to fit a velocity reference. This version can have advantages if an accurate velocity reference (e.g. Doppler laser) is used instead of an acceleration reference. The observation equation can be rewritten as follows.

$$\int(c_0 + c_1 y(t) + c_2 y(t)^2 + c_3 y(t)^3 \dots)dt + \int(b_1 t + b_2 t^2 + b_3 t^3 \dots)dt + v_0 = v_{ref} \quad (4.23)$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots \\ 1 & \int_{t_0}^{t_1} 1\,dt & \int_{t_0}^{t_1} y(t)\,dt & \int_{t_0}^{t_1} y(t)^2\,dt & \int_{t_0}^{t_1} y(t)^3\,dt & \dots & \int_{t_0}^{t_1} t\,dt & \int_{t_0}^{t_1} t^2\,dt & \int_{t_0}^{1} t^3\,dt & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & \int_{t_0}^{t_n} 1\,dt & \int_{t_0}^{t_n} y(t)\,dt & \int_{t_0}^{t_n} y(t)^2\,dt & \int_{t_0}^{t_n} y(t)^3\,dt & \dots & \int_{t_0}^{t_n} t\,dt & \int_{t_0}^{t_n} t^2\,dt & \int_{t_0}^{t_n} t^3\,dt & \dots \end{bmatrix} \begin{bmatrix} v_0 \\ c_0 \\ c_1 \\ c_2 \\ c_3 \\ \dots \\ b_1 \\ b_2 \\ b_3 \\ \dots \end{bmatrix} = \begin{bmatrix} v_{ref}(t_0) \\ v_{ref}(t_1) \\ \dots \\ v_{ref}(t_n) \end{bmatrix} \quad (4.24)$$

In these formulations for least squares the bias drift has been approximated as a time varying polynomial. The choice of the polynomials order should be based on the variability of the bias and the length of the signal. Practically, the limit is the computing precision. From experimentation with this formulation, tests have shown polynomials with order greater than 15 cause the Gramian matrix to become ill conditioned and near singular, which can be attributed to the finite precision computations. This causes a loss of accuracy and eventually results in meaningless solutions. The recommendation is that the least

squares solution for this formulation should only be used with lower order bias drift functions and for short data sets. For longer data sets the recommendation is to use a Kalman filtering approach described in the next section.

4.3.2 Kalman Filter

While least squares provides a quick solution for short data sets, some problems exist in the determination of an appropriate order for the bias drift function. As the data set becomes longer or the variations more frequent the polynomial order must become increasingly large to properly describe it.

An alternative approach for estimating the compensating polynomial is to use a Kalman filter, which allows for a bias model which is not limited by order. The requirement is that the problem must be changed so that the coefficients are arranged as elements of the state vector. The first step is to define the dynamics model starting with the following equation,

$$\dot{v}(t) = c_0 + c_1 y(t) + c_2 y(t)^2 + c_3 y(t)^3 + c_4 y(t)^4 + c_5 y(t)^5 \qquad (4.25)$$

If all the terms of the above equation are assumed to be "random constants", a dynamics model can be defined as in equation (4.27). However consideration must be made for the additional dynamics which will be present in the bias drift c_0. This parameter is modeled as a random walk. Under the linear formulation of the Kalman filter the dynamics matrix, $\boldsymbol{F}$, is assumed to be linear time invariant. The following formulation maintains the assumption of the linearity as no linearization is required to write the matrix, however the matrix is not time invariant. As a result the state transition matrix is recalculated each time step using the 2nd order Taylor series approximation.

$$\dot{x} = \boldsymbol{F}(t)x + \boldsymbol{Q}_k \tag{4.26}$$

$$\begin{bmatrix} \dot{v} \\ \dot{c_0} \\ \dot{c_1} \\ \dot{c_2} \\ \dot{c_3} \\ \dot{c_4} \\ \dot{c_5} \\ \dot{b_{rw}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & y(t) & y(t)^2 & y(t)^3 & y(t)^4 & y(t)^5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ b_{rw} \end{bmatrix} + \boldsymbol{Q}_k \tag{4.27}$$

$$\boldsymbol{\Phi}_{k,k+1} = \begin{bmatrix} 1 & t+\frac{t^2}{2} & \left(t+\frac{t^2}{2}\right)y(t) & \left(t+\frac{t^2}{2}\right)y(t)^2 & \left(t+\frac{t^2}{2}\right)y(t)^3 & \left(t+\frac{t^2}{2}\right)y(t)^4 & \left(t+\frac{t^2}{2}\right)y(t)^5 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \left(t+\frac{t^2}{2}\right) \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.28}$$

The use of numerical approximation for the state transition matrix is analyzed to determine the percent error for a given step size. The errors are compared for one of the terms $e^{\Delta t}$.

Table 4.2 Numerical Approximation Errors in Individual Steps

% error	Δt=1	Δt=0.1	Δt=0.01	Δt=0.001	Δt=0.0001	Δt=0.00002
$1+\Delta t$	26.4	0.467	4.96e-3	4.99e-5	5e-7	1.99e-8
$1+\Delta t+\frac{\Delta t^2}{2}$	8.03	0.0154	1.65e-5	1.66e-8	1.66e-11	1.33e-13
$1+\Delta t+\frac{\Delta t^2}{2}+\frac{\Delta t^3}{6}$	1.89	3.84e-4	4.13e-8	4.17e-12	<2e-16	<2e-16

Table 4.2 illustrates a relationship between the time step size and the approximation order required to maintain accuracy. For large step sizes, a high order approximation will be

required to maintain accuracy. Due to the increasing number of operations when using smaller step sizes, the accumulation of error must also be considered.

Table 4.3 Numerical Approximation Errors Accumulated For 1 Second Interval

% error	Δt=1	Δt=0.1	Δt=0.01	Δt=0.001	Δt=0.0001	Δt=0.00002
$1+\Delta t$	26.4	4.67	4.96e-1	4.99e-2	5e-3	5e-4
$1+\Delta t+\frac{\Delta t^2}{2}$	8.03	0.154	1.65e-3	1.66e-5	1.66e-8	6.65-9
$1+\Delta t+\frac{\Delta t^2}{2}+\frac{\Delta t^3}{6}$	1.89	3.84e-3	4.13e-6	4.17e-9	<2e-12	<2e-12

This result exemplifies a general trend seen throughout this thesis. As the step size is reduced it is possible to use lower order approximations with similar results. However in the area of post processing (not real-time), it is possible to use higher order approximations without significant impact on time of computation. Just for record, the data I collect is sampled at above 10 kHz and the Kalman filter is updated at every sample.

The measurement model to introduce the laser reference data is simple. The laser reference is equal to the first state in the state vector. Thus the measurement matrix is a unit vector with the first element 1.

$$\boldsymbol{H} = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0] \qquad (4.29)$$

To illustrate the results with the Kalman filter a simulated burst series is generated using the accelerometer simulator. The series is shown in Figure 4.12 with a subsection showing that each burst is a 1 second duration, 10 Hz sinusoid. This signal is chosen as it will illustrate the apparent bias shift when not corrected.

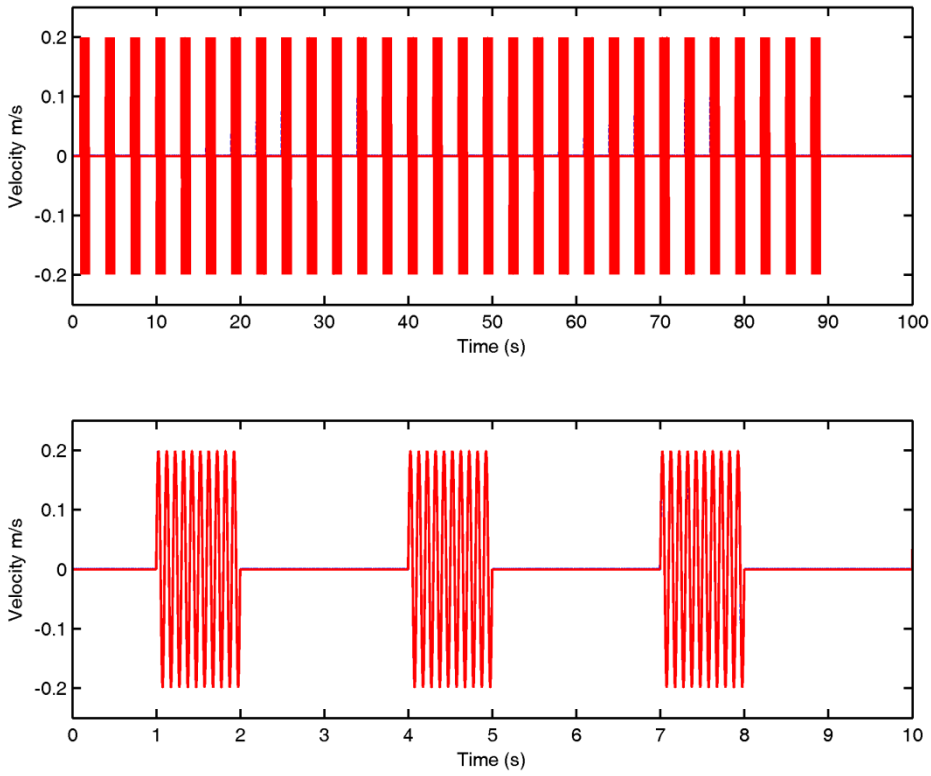

Figure 4.12 Kalman Filter Simulated Input

To improve the speed of convergence of the Kalman filter, a least squares estimate is used to initialize the state vector.

To verify that the Kalman filter converges given an initial error in the state estimate, an initialization error has been introduced into the C2 term which results in vibration rectification error during oscillation. As the Kalman filter converges the pulsation of VRE error in the C0 term is reduced indicating that the solution has improved.

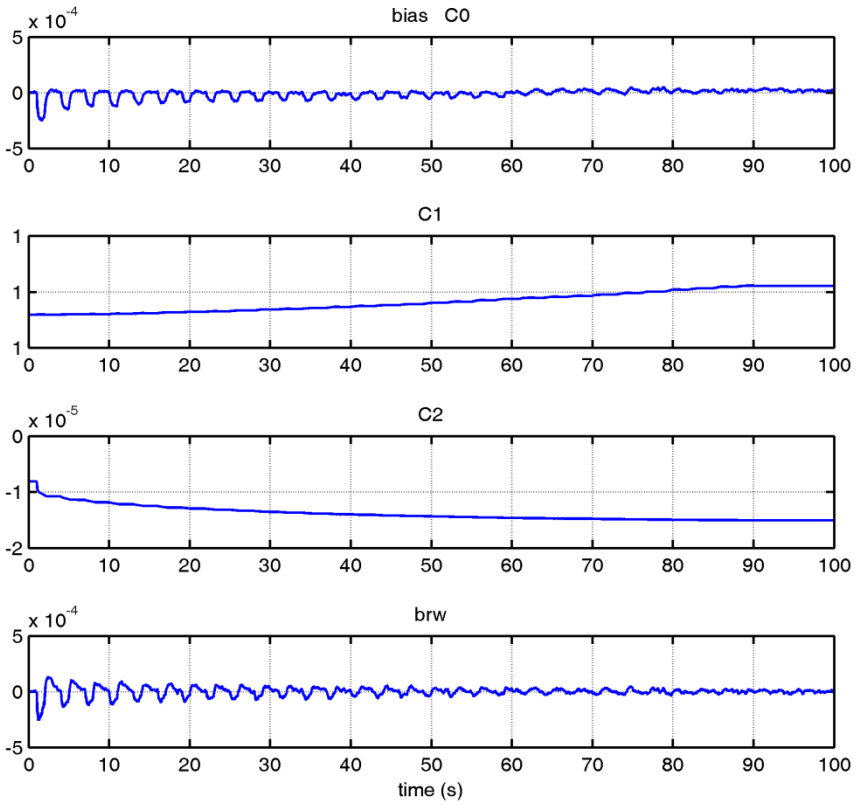

Figure 4.13 Convergence of Kalman Filter Estimates in the Case of Poor Initial Estimate

When a better initial estimate is given, as in Figure 4.14, the states change very little from the initial. The C0 term appears to be constant and appears not to contain any resemblance of the vibration rectification. However, it contains a visible bias drift.

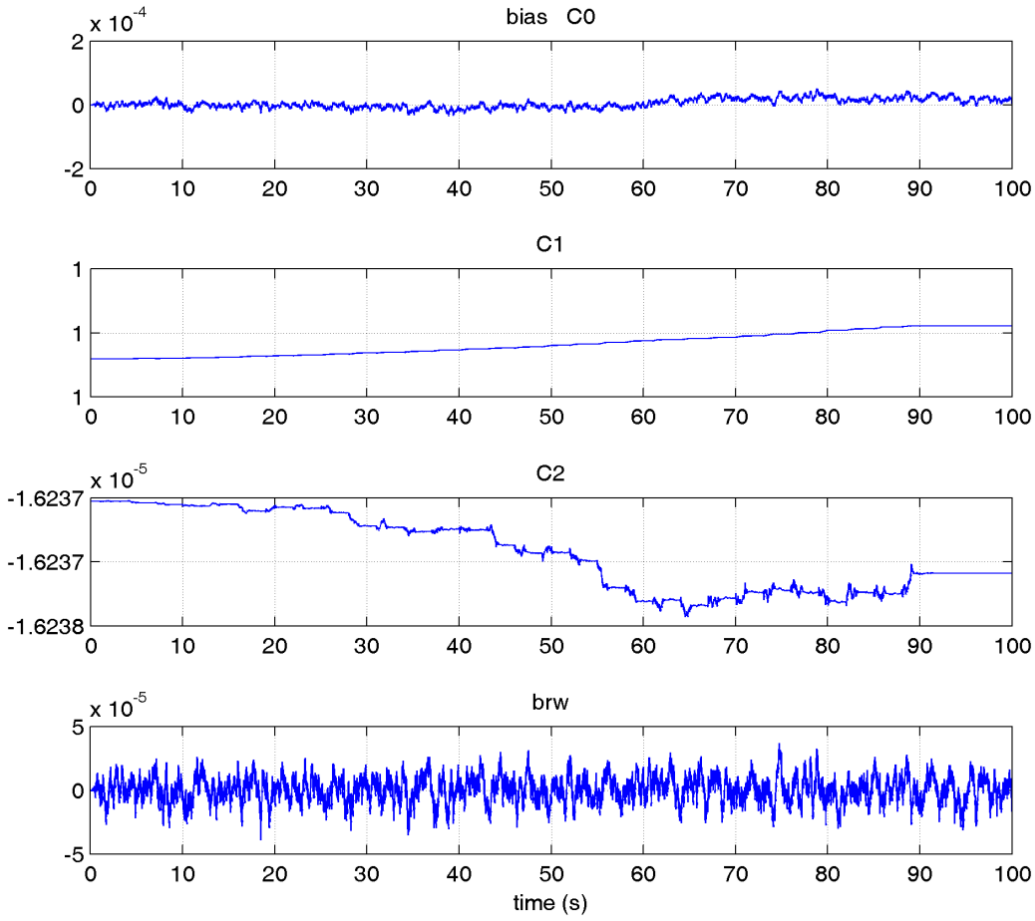

Figure 4.14 Kalman Filter Results with Good Initial Estimate

The analysis of residuals is used to determine if and when the Kalman filter has converged as well as the level of fit. The residual in Kalman filtering is the innovation matrix v which defines the difference between the current estimate and the measurement.

$$v_k = z_k - \boldsymbol{H}_k \hat{x}_k^- \qquad (4.30)$$

Analysis of the frequency content of this signal can determine if the filter has converged. If the filter converges and the solution is a good fit, only white noise is left in the innovation sequence. Figure 4.15 shows the frequency spectrums of the innovation sequence for various levels of the convergence starting from the zero initial condition. The filter start

and most of the midway results show clear indications of residual harmonics. The final result shows a nearly white spectrum indicating that the filter has converged and fit well. Just for reference the least squares initialization will start the KF at what is shown here as KF end.

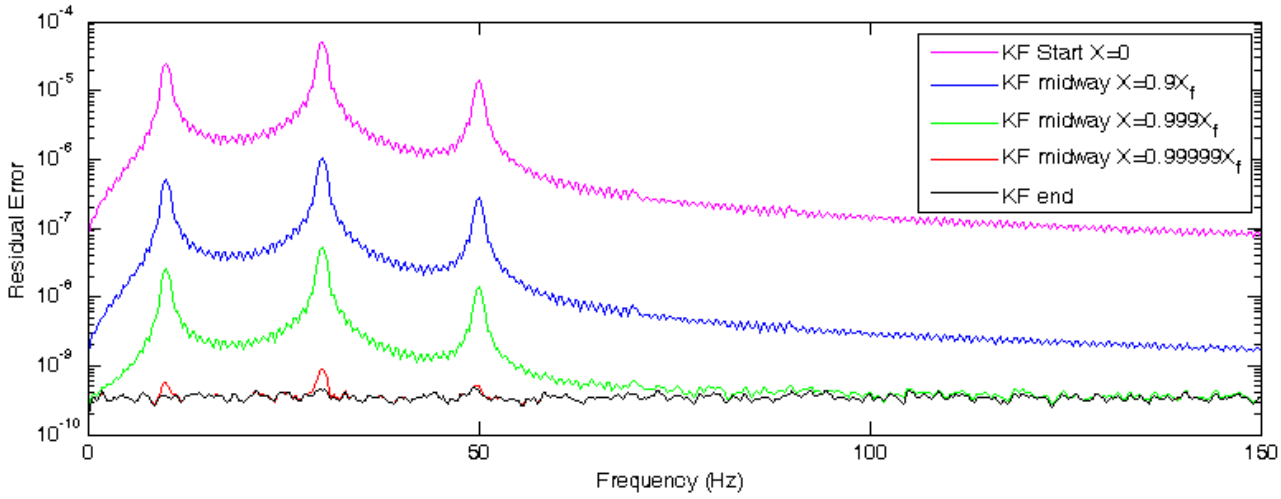

Figure 4.15 FFT of Kalman Filter Residuals

While testing the innovation sequence is usually enough to determine if the Kalman filter has functioned. For the purposes of determining if the polynomial model is correct it is also necessary to evaluate the bias for the same results. During the operation of the filter a certain amount of error is attributed to the bias. This is to allow it to drift in response to the accelerometer drift but also to allow for some modeling errors. In this model, most of the modeling errors, which are large when the filter has not yet converged, get added to the bias term. As a result the bias term absorbs some of the remnants of the uncorrected harmonics. Setting the process noise for the bias too small will prevent convergence altogether as the filter is unable to immediately change the other coefficients to eliminate the error. When converged, the bias should be reduced to a nearly white noise spectrum with no visible harmonics from the excitation as it does in Figure 4.16.

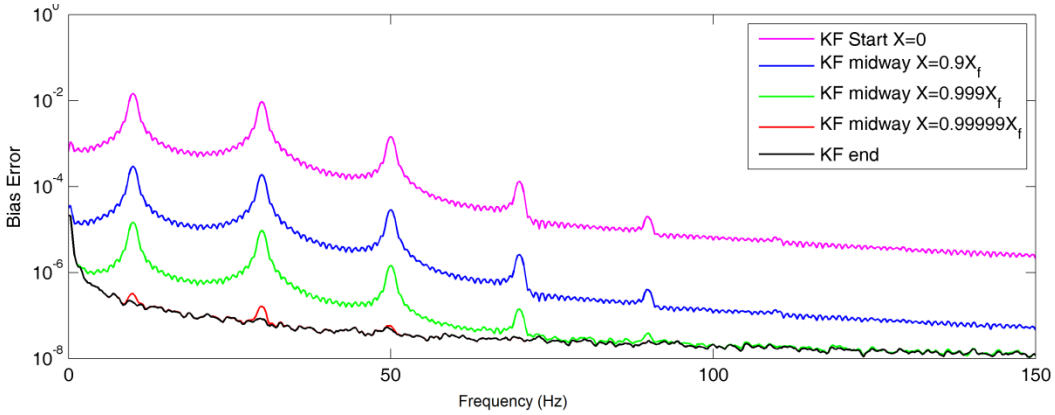

Figure 4.16 FFT of Kalman Filter Bias Noise, C0 term

This Kalman filter formulation, while essentially performing the same task as the least squares, has a significantly longer convergence time and more data is required. As a result it is not as applicable for the short burst testing. It does however have the potential use in tracking changes in nonlinearity over time or through various temperature ranges.

4.3.3 Determining Polynomial Order

When using a polynomial representation of nonlinearity a key question is always asked: what order of polynomial is required to correct the nonlinearity? This question can only be answered in terms of an acceptable error. Since the inverse polynomial is an approximation, the higher the order the better the approximation. High order polynomials are however undesirable due to computational time and precision so a compromise must be made based on the minimum acceptable error.

The first test is in the estimation procedure. It was mentioned in the Kalman filtering example that the residual and bias can be examined to determine if harmonics were left. If these harmonics are significant (above the noise floor), this would be an indication that the polynomial has an insufficient order.

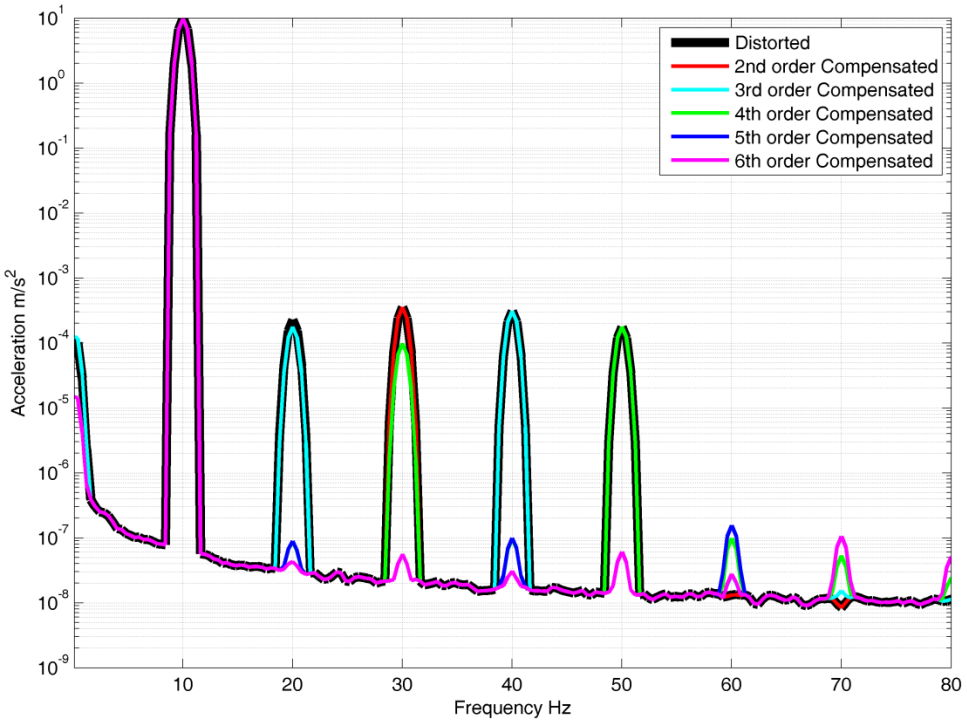

Figure 4.17 Residual Harmonics for Various Compensating Polynomials

The second test is to directly test the vibration rectification for various amplitudes and frequencies using a given polynomial order. Keep redoing the test with increasing order until a plateau is reached where the error is acceptable or not further correctable.

4.4 Conclusions

This chapter has described a testing method for accelerometers which allows the entire nonlinearity to be identified over a short time period. This method is useful as it limits the amount of self-heating thus allowing better separation of the temperature and acceleration related nonlinearities. This chapter has also presented estimation methods suitable for this type of calibration.

CHAPTER 5: IMU SYSTEM DESIGN

Through identifying some of the major contributors to the sensor error, a set of specifications can be generated for a system designed for high vibration and precision inertial measurement. These design requirements are independent of the sensor types used.

Table 5.1 Hardware Design Requirements

Error	Requirements for resolution
Vibration rectification	• High sample rate • Maintain broad bandwidth prior to compensation • High processing throughput • Vibration damping shock absorbers if applicable to reduce bandwidth of vibration prior to sensors
Noise	• Low jitter oscillator for ADC • Low noise amplifiers and ADC • Low Noise Power Supply
Drift	• Precision resistors and capacitors • Low self-heating • Low drift amplifiers • Low drift power supply • Monitoring the temperature of sensors, amplifiers, and ADCs
Delay and Synchronization	• Minimum necessary filtering to reduce delays • Precision timing of samples • Higher sample rates to minimize sample size • Simultaneous sampling ADC or similarly synchronized

The priorities of reducing size, cost and power have meant that many inertial systems are not capable of meeting all of these requirements. Some other constraints have been the availability of devices and components. Only recently has the technology for ADCs and FPGAs made it possible to achieve high sample rates with high resolution converters. This is coupled with many other improvements in electronic component designs which lower power consumption without significant increases in noise or cost.

5.1 Inertial Measurement Unit

The data acquisition system is essentially all components required to take raw measurements from the sensors and convert them to position, velocity and attitude. This process includes signal processing, data conversion, timing, compensation, mechanization and communication. A sensor cluster was designed using the QA700 accelerometers. The superior linearity make them the best choice even given the ability to compensate.

5.1.1 Data Acquisition Hardware

The data acquisition hardware is essentially motherboard which provides signal and power routing to the FPGA board as well as to a set of ADCs. The FPGA essentially has dedicated resources for each device so the addition of other sensors does not impact the timing or the ability for the data to be processed. In this way, this system is easily extensible. To test concepts of the design, a prototype, laboratory grade data acquisition system was assembled using the TI ADS1278 ADC [25]and an Opal Kelly XEM6001 FPGA board [29]. This was used to determine system noise, drift, sample rate requirements, and analyze timing and delays. The test results were used to develop a more complete system which used the enhanced FPGA, model Opal Kelly XEM6010-LX150 [30].

Figure 5.1 IMU Data Aquisition System

5.1.2 Implementation of Mechanization in FPGA

Implementing the inertial mechanization in a strong vibratory or high dynamic environment has high demands on processing power. As mentioned previously in this thesis, one approach to maximize performance is to implement a multi-rate processing technique. The technique discussed in [4] uses 3 different rates:

- J cycle: Very High Speed used for computation of the rotation vector from angular rates and integration of the specific force. This cycle speed is also used for application of corrections to sensors.
- K cycle: Reasonably Fast used for attitude update, Velocity and position updating
- I cycle: Slow Speed used for navigation frame rotation update as well as coriolis and gravity calculation

In most systems all these operations take place in a single processor or a few processors. The disadvantage of this solution is that processes are not independent. The completion of one process is required before the next can start. The architecture of an FPGA allows for

the implementation of both parallel and sequential processing algorithms. The key advantage for this solution will be the implementation of the j cycle algorithms in direct hardware. Instead of using a processor, the compensation, integration, and other low level operations can also be implemented using FPGA Floating Point cores and state machines. A comparison of representative processing capability between a DSP and FPGA is shown below.

Table 5.2 Computation Comparison Single Precision Multiplications Additions

Device	Computation Rate, Single Precision multiplications or additions per second	Device Utilization Percentage of Time or Logic Used
DSP	37,000,000	100% of DSP
Microblaze Softcore Processor	16,000,000	100% of Core 5% of FPGA Logic Spartan 6 LX150
Dedicated Multiplier/Adder Core	200,000,000	0.2% of FPGA Logic Spartan 6 LX150

Table 5.3 Computation Comparison Single Precision Divisions

Device	Computation Rate, Single Precision divisions per second	Device Utilization Percentage of Time or Logic Used
DSP	1,000,000	100% of DSP (Division not directly supported by instructions)
Microblaze Softcore Processor	2,500,000	100% of Core 5% of FPGA Logic Spartan 6 LX150
Dedicated Multiplier Core	200,000,000	0.7% of FPGA Logic Spartan 6 LX150

The computation comparison table clearly demonstrates that the replacement of a DSP with the softcore processor would result in a large reduction in computation capacity for multiplication and addition but not for division which is directly supported by the softcore.

However the use of dedicated math operators in the FPGA logic would see a massive increase in potential processing capability. It should be possible to implement the relatively simple high speed j-cycle operations as well as the sensor compensation algorithms using these dedicated operators. The lower speed k and i cycle operations can then be performed by one or multiple soft-core processors. This is represented schematically in the block diagram on Figure 5.2.

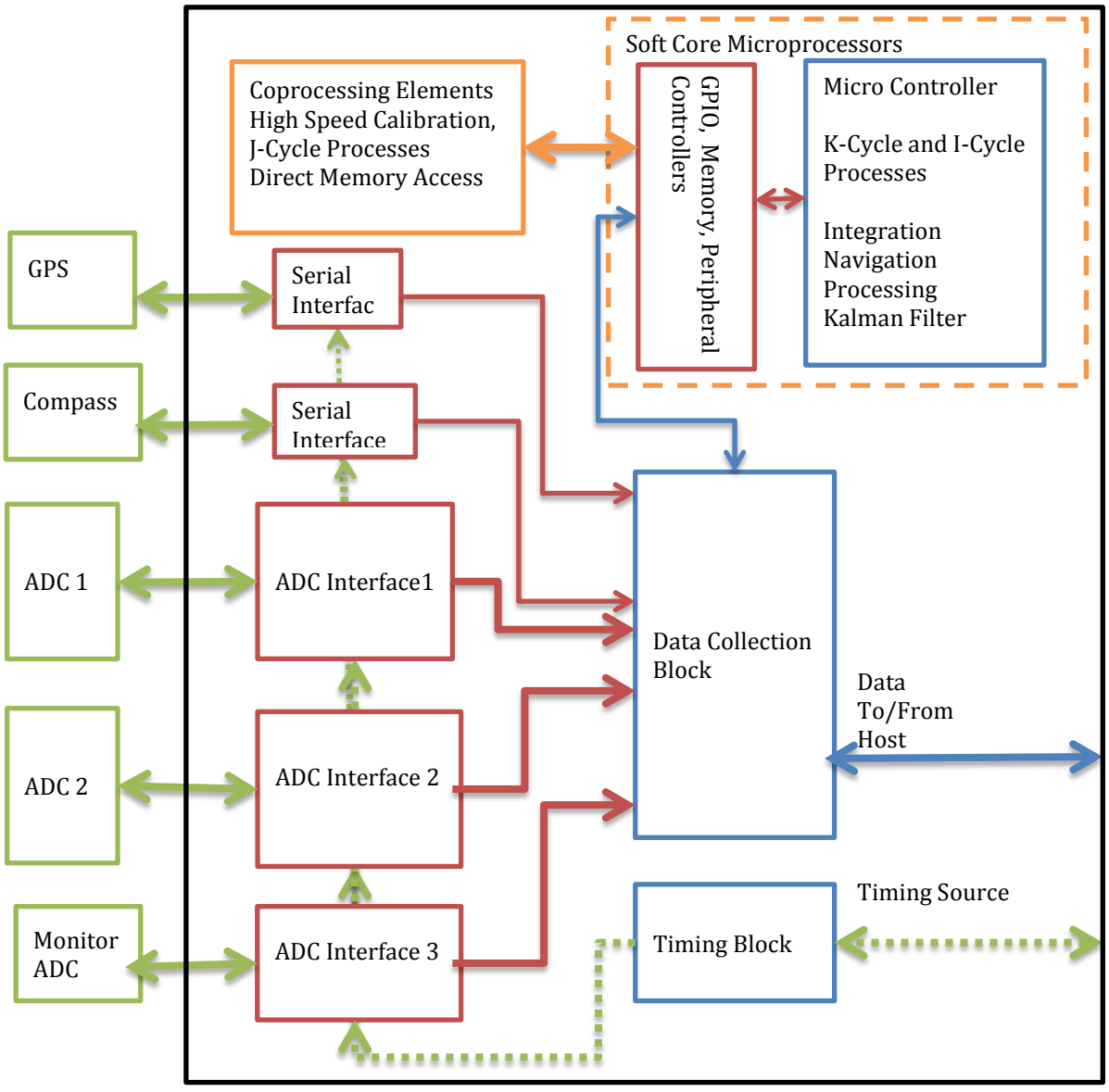

Figure 5.2 FPGA IMU Data Acquisition and Processing Architecture

5.2 Power Supply

The development of a precision power supply was one of the first tasks of the design process. It was clearly stated in the datasheet for the accelerometers and gyros as well as the amplifiers and other parts that the scale factor, bias and various other factors would be affected by changes in power supply voltage. Tests were performed to identify the sensitivity of standard linear regulators to the load current change. It was found that a change of almost 3% of the nominal output would occur over a 0.5 A range. This is compounded by the voltage losses due to path resistance. To minimize the errors associated with these variations, the power supply was designed with precision power op amps intended for lab power supplies [31]. The output voltage was sensed at the sensor cluster to minimize the path resistance effects. This was combined with a precision reference to make an output voltage that changed no more than 10 μV over a change of 0.5 A of current draw, a 3 order of magnitude improvement.

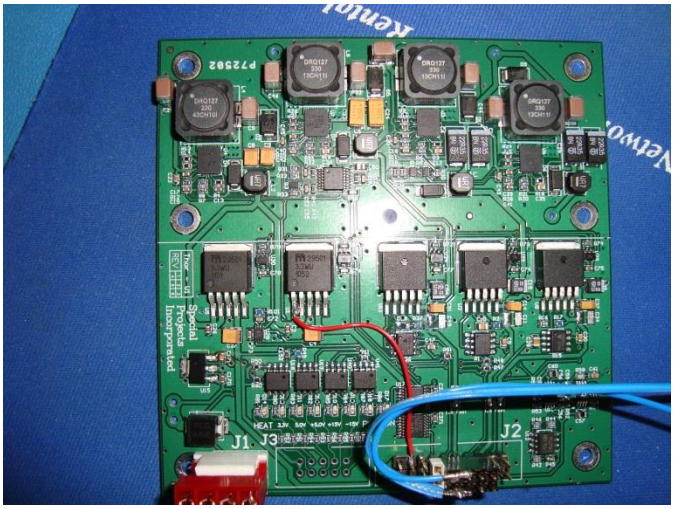

Figure 5.3 IMU Power Supply

5.3 Sensor Cluster

The sensor cluster for this system was designed to support a set of 3 Honeywell QA700 accelerometers and 3 Systron Donner LCG-50 gyros [32]. The system could easily be reconfigured for different types of sensors as the data acquisition system has universal inputs. The QA series accelerometer uses a scaling resistor to set the feedback loop strength and also the output scaling. To reduce self-heating and improve accuracy for the smaller dynamic range, the scaling resistor was chosen to allow for $\pm 4g$.

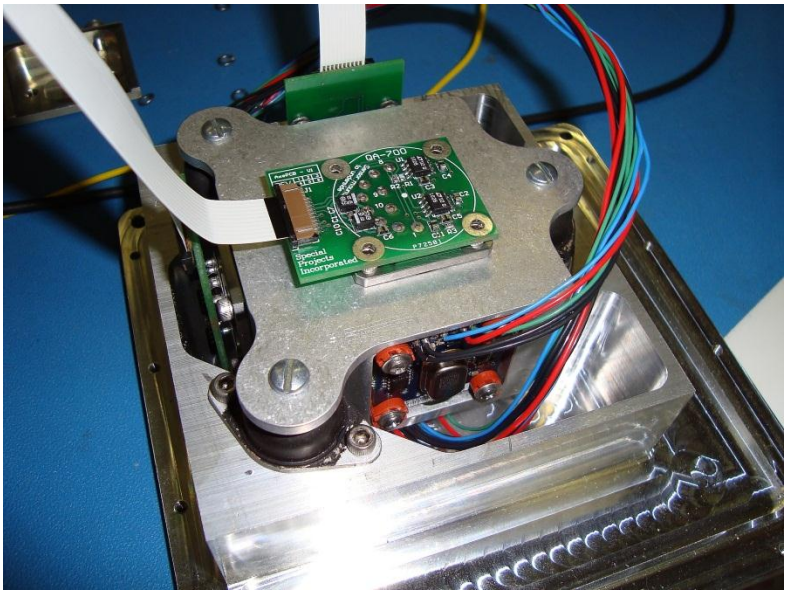

Figure 5.4 IMU Sensor Cluster

5.4 System Testing

While the system has not been tested in a navigation setting, many of its components have been evaluated using the experimental facility at the University of Calgary. Parts of this system were used to generate many of the experimental results already shown.

During lab testing the cluster was used with and without the shock absorbers. The results indicate that the absorbers effectively reduce the high frequency vibrations but are highly

nonlinear and non-symmetrical. This would not normally be a concern unless the system was required to accurately measure the movements of the base or was referenced to the base as it is in system testing. For these tests the mounts were replaced with hard mounting blocks to prevent any movement between the cluster and base.

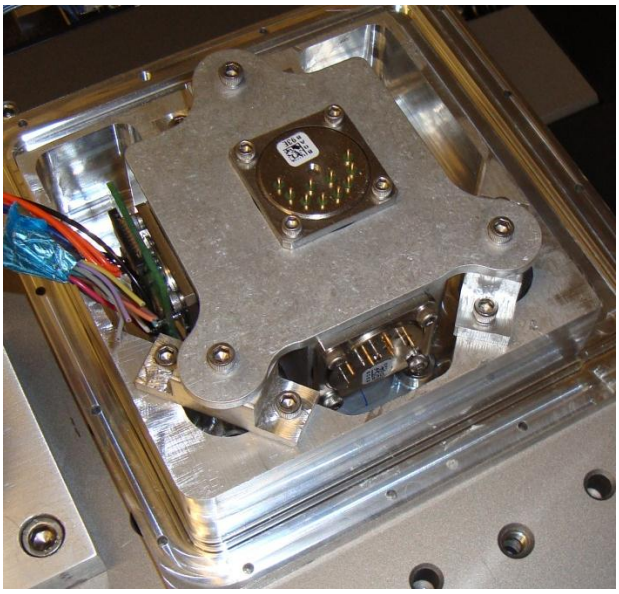

Figure 5.5 Cluster Hard Mounted

5.5 Conclusions

The system described above was designed during this research to fulfill the requirements of precision measurement under strong vibration. At the time the thesis was written the system had not yet been completed. Specifically the mechanization had not been implemented. The system designed has the following features.

- Accelerometers and Gyroscopes sampled with 24bit ADC at 50 kHz
- Non-linearity compensation at 50 kHz using hardcoded FPGA Logic
- Temperature measurement using internal and external temperature sensors on accelerometers and gyros.

- Xilinx Spartan-6 LX150 FPGA for data collection and real time processing
    - Multiple soft core microprocessors
    - Dedicated logic for high speed repetative calculations
- Highly modular, expandable, and reconfigurable.
- Availability of all raw data to a host computer through USB

CHAPTER 6: CONCLUSIONS AND RECOMENDATIONS

The primary objective of the research described in this thesis is to improve the accuracy of inertial measurement under vibration. The focus of this objective was vibration rectification error (VRE). This thesis has redefined VRE to describe a number of vibration related errors. It has demonstrated the major source of VRE as the sensor's nonlinearity but also some less obvious sources, such as the numerical integration errors, self-heating and mechanization. This thesis has also sought to provide solutions to these problems through analysis, estimation and testing methods.

The investigations have shown that many vibration related issued can be directly compensated or avoided by using high sample rates. To this end a system was designed which can consistently operate at 50 kHz sample rate. This is achieved using an advanced ADC coupled with the use of an FPGA based data acquisition and processing unit. A processing architecture was proposed for this system which will allow for computations to be performed at high rate, potentially allowing the system to completely measure the vibrations and high dynamics up to approximately 10 kHz. Many other errors can be eliminated by employing additional sensors. In particular measuring the temperature of all inertial sensors and acquisition components will allow for compensation of thermal related errors.

Specific contributions

The major contributions of this research include:

a. The definition and analysis of errors which impact the accuracy of the IMU when under vibration. These include, VRE due to nonlinearity, errors in integration, self-heating,

timing or signal delay, and mechanization related errors. The relation of these errors to sensor specifications has been identified. The key parameters identified are nonlinearity, bias stability and temperature sensitivity. These parameters are tested for a selection of accelerometers. The relative magnitudes of the errors are calculated.

b. Analysis and recommendations for the compensation of vibration related errors. This research has identified a direct link between the sample rate and the accuracy under vibration.

- Reducing errors due to numerical integration.
- Providing sufficient bandwidth to measure vibration so that non-linearity and thus VRE can be compensated
- Reducing errors in mechanization due to finite series approximations
- Reducing delay and time synchronization errors

These effects are rarely considered independently. The example of section 3.3.2 showed that when combining the nonlinearity compensation and integration, a sample rate of 1 kHz was the minimum requirement for a vibration of 10 Hz. This indicates the need to sample significantly faster than the 100 Hz normally used of inertial systems.

Another link has been made to vibration related self-heating. The error is not generally considered vibration rectification however the result is the same. Vibration may cause self-heating of the sensor. This heating manifests as a DC output when combined with the temperature sensitivities. To compensate this error, the temperature of the sensitive element must be known. Limitations on the placement of temperature sensors necessitate prediction of the temperature rise based on acceleration.

c. A calibration method has been identified which enables the parameters of the non-linearity to be estimated while minimizing the effects of self-heating. A method based on the vibration testing, but utilizing a linear stage, is used to identify the nonlinearity for both positive and negative accelerations. Results from this thesis have indicated that this method of testing can estimate the nonlinearity in 10 seconds sufficient to reduce VRE 10 to 1000 times depending on the initial level of nonlinearity. In most cases the VRE could be reduced to below the bias drift level. This method has the advantage of being able to test a larger range of acceleration and broader frequency range than the commonly used tilt table. It also allows both directions to be tested simultaneously.
d. Requirements for IMU system design are given based on the research and testing. Along with these, a system has been developed which meets the requirements. Among the key objectives was achieving the high sample rates and processing rates. An ADC was selected allowing for continuous simultaneous sampling of 8 channels at 50 kHz in high, 24 bits resolution. To accommodate the data collection rates and achieve the timing accuracy, an FPGA is used instead of a microcontroller.

Recommendations

This work has brought to light a number of important topics which could not be fully tested within this MSc research. Among them is the verification of self-heating by accurate temperature measurement. The degree of self-heating on sensors such as the MEMS PCB accelerometer is small, however the sensitivity of bias drift due to temperature is high. It was determined that a measurement accuracy of 0.001°C was required to properly compensate the error. To obtain measurements on that order special equipment was

required, which was developed in this research. It is readily available to study and characterize the power dissipation vs acceleration.

Another issue is one of power consumption. This is a concern with regard to the self-heating of IMU as a whole. Some areas have already been identified where power can be conserved. Work is ongoing to reduce this further. A long term option would be lower power FPGAs. Xilinx currently has lower power devices and is continually improving them. The currently used FPGA should be swapped out for a lower power version when possible.

This research focused on acquisition of signals and suppression of the effects of sensor imperfections. It will be the task of future research to implement full inertial mechanization and conduct complete system test in various environments. It is also recommended to continue the research to determine if other models can be used to more suitably represent the sensor nonlinearities.

REFERENCES

[1] IEEE Std 1293-1998, IEEE Standard Specification Format Guide and test Procedure for Linear, Single Axis, nongyroscopic Accelerometers, IEEE, 1998.

[2] J. Farrel, GNSS Aided Navigation and Tracking, Baltimore: American Literary Press, 2007.

[3] R. Dorobantu and B. Zebhauser, Field Evaluation of a Low-Cost Strapdown IMU by means of GPS, Munchen, Germany.

[4] D. Titterton and J. Weston, Strapdown Inertial Navigation Technology 2nd Ed, Institution of Electrical Engineers, 2004.

[5] P. Savage, Strapdown Analytics, Maple Plain, Minnesota: Strapdown Associates Inc., 2000.

[6] A. Chatfield, Fundamentals of High Accuracy Inertial Navigation, American Institute of Aeronautics, 1997.

[7] Barbour and Schmidt, "Inertial Sensor Technology Trends," *IEEE Sensors Journal,* pp. Vol. 1, No. 4 2001 pp 332-339, 2001.

[8] S. Spiewak and S. Ludwick, "Model Based Compensation of Vibration Rectification in Precision Inertial Sensors," in *Euspen International conference*, San Sebastian, 2009.

[9] E. Jacobs, "New Developments in Servo Accelerometers," *Institute of Environmental*

Sciences 14th Annual Technical Meeting, 1968.

[10] J. Bortz, A new mathematical formulation for strapdown inertial navigation, IEEE Transactions on Aerospace and Electronic Systems, 1971.

[11] R. Rogers, Applied Mathematics in Integrated Navigation Systems, Third Edition, American Institute of Aeronautics, 2007.

[12] A. M. Bruton, "Improving the Accuracy and Resolution of SINS/DGPS Airborne Gravimetry," Geomatics Engineering, University of Calgary, Calgary, 2000.

[13] C. Glennie, "An Analysis of Airborne Gravity by Strapdown INS/DGPS," Geomatics Engineering, University of Calgary, 1999.

[14] C. Naranjo, Analysis and Modeling of MEMS based Inertial Sensors, Stockholm: School of Electrical Engineering, Kungliga Tekniska Hgskolan, 2008.

[15] S. Zhang, "Characterization of Nonlinearity Using Polynomial Transfer Function of All-Optical Sampling," *Microwave and Optical Technology Letters, Vol 53, No 1,* 2011.

[16] IEEE Std 528-2001, IEEE Standards for Inertial Sensor Terminology, IEEE, 2001.

[17] Honeywell, "QA700 Datasheet," [Online]. Available: http://www51.honeywell.com/aero/common/documents/myaerospacecatalog-documents/Missiles-Munitions-2/QA700_DataSheet.pdf.

[18] PCB Piezotronics, "85_Model 3701G2FA3G Capacitive Accelerometer.pdf," [Online].

[19] Kistler, "84_8330.pdf," [Online].

[20] IEEE Std 836-1991, IEEE Recommended Practice for Precision Centrifugre Testing of Linear Accelerometers, IEEE, 1992.

[21] Colibrys, SF3000L Datasheet.

[22] Honeywell, QA2000 datasheet.

[23] R. Moghimi, Understanding Noise Optimization in Sensor Signal-Conditioning Circuits, Analog Devices Inc., 2008.

[24] R. Reeder, Analog-to-Digital Converted Clock Optimization, Analog Dialogue 42-02, February, 2008.

[25] Texas Instruments, ADS1274 ADS1278 Datasheet, Quad/Octal Simultaneous Sampling, 24-Bit ADC, 2007.

[26] B. Carter, Op Amp Noise Theory and Applications, Excepted from Op Amps for Everyone, Texas Instruments.

[27] M. Grewal and A. Andrews, Kalman Filtering Theory and Practice Using Matlab, 2nd Ed, Wiley, 2001.

[28] P. a. W. Franklin, Digital Control of Dynamic Systems 3rd Ed, Ellis-Kagle Press, 1997.

[29] Opal Kelly, http://www.opalkelly.com/library/XEM6001-UM.pdf.

[30] Opal Kelly, http://www.opalkelly.com/library/XEM6010-UM.pdf.

[31] Texas Instruments, www.ti.com/lit/ds/symlink/opa544.pdf.

[32] Systron Donner, http://www.systron.com/products/lcg50.

[33] A. Gelb, Applied Optimal Estimation, The MIT Press, Massachusetts Institute of Technology, 1974.

[34] T. Weise, Global Optimization Algorithms Theory and Application 2nd ed, 2009.

[35] L. Christel and M. Bernstein, "Vibration Rectification in Siliicon Micromachined Accelerometers," *Solid-State Sensor and Actuators, Dgiest of Technical Papers,* 1991.

[36] C. Zaiss and S. Spiewak, "Vibration Rectification and Thermal Disturbances in Ultra Precision Inertial Sensors," *ASME IMECE,* 2011.

[37] S. Pethel, Test and Evaluation of High Performance Micro Electro-Mechanical System Based Inertial Measurement Units, Navigation Technology Associates.

[38] G. Schmidt, INS/GPS Technology Trends, NATO RTO Lecture Series, 2011.

[39] P. LaFond, Modeling for Error Reduction in vibrating Beam Accelerometers, Sundstrand Data Control.

[40] IEEE Std 952-1997, IEEE Standard Specification Format Guide and Test Procedure for Single Axis Interferometric Fiber Optic, IEEE, 1997.

[41] Foote and Grindeland, "Model QA3000 Q-FLEX Accelerometer High Performance Test Results," *IEEE Aerospace and Electronics Systems Magazine, Volume 7 Issue 6,* pp. 59-67, 1992.

[42] Chapra, Numerical Methods for Engineers, 4th Ed, McGraw Hill, 2002.

[43] Texas Instruments, OPA1632 Datasheet, 2010.

Appendix A

Accelerometer Simulator

During the development of some of these mathematical techniques and for verification of error behaviors an accelerometer simulator was developed. This model includes nonlinearity as well as random walk bias shift. It was used to verify the least squares and kalman filter estimation techniques as well as to predict the expected behavior of an accelerometer to a particular type of test or analysis. The following equations describe the simulators behavior.

Firstly this simulator is a simple input output model as seen in equation A.1 . Its input is the true acceleration x; its output y(x) is a signal which imitates an accelerometer output with a nonlinearity.

$$y(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 + \cdots + bias + noise \qquad (A.1)$$

The terms a_0 to a_n represent the nonlinearity. The bias term is the random bias drift and is modeled as a random walk which is by definition the integral of a random sequence. Equation 3.35 describes this integral where w(t) is zero mean white noise.

$$bias = \int w(t)\, dt \qquad (A.2)$$

To add some extra realism some extra noise is added to simulate quantization and intrinsic noise. This is done simply by adding some more white noise q(t).

$$noise = q(t) \qquad (A.3)$$

The overall simulator model is as follows.

$$y(t) = a_0 + a_1x(t) + a_2x(t)^2 + a_3x(t)^3 + a_4x(t)^4 + a_5x(t)^5 + \cdots + \int w(t)\,dt + q(t) \quad (A.4)$$

Where a_0 to a_n, w(t) and q(t) are simulated model parameters, x(t) is the input acceleration, and y(t) is the output signal.

As a test of the approximated bias drift and noise performance the Allan variance can be produced for the simulated data and compared to a real accelerometer. As you can see from Figure A., the plot for the simulator generated output has similar quantization noise, bias instability and random walk drift to that of a QA700 shown in Figure 3.7.

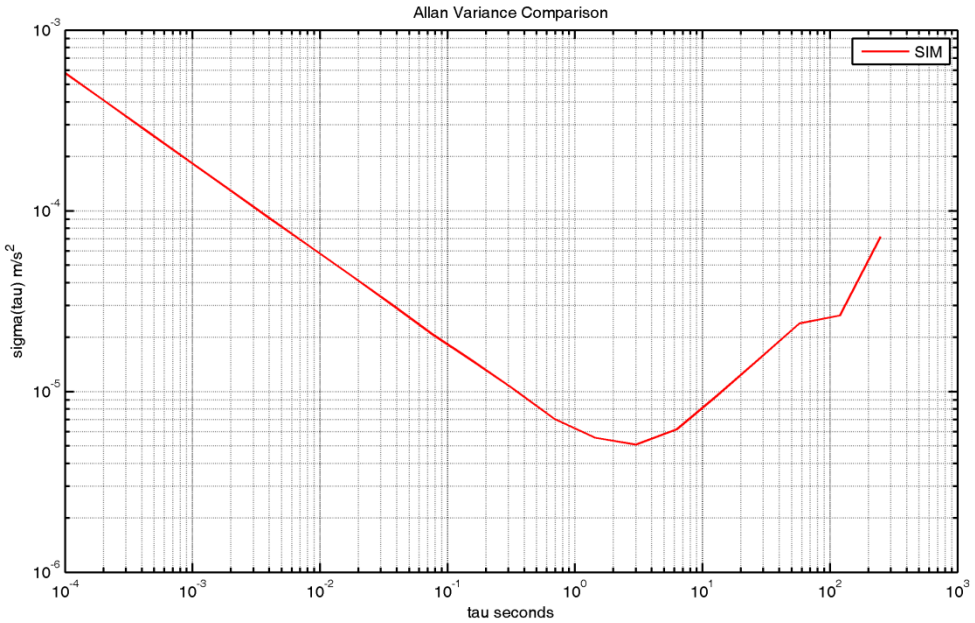

Figure A.1 Allan Variance of Simulated Accel

Appendix B FPGA's Explained

In a standard microcontroller a processor is attached to peripherals and memory. Operations are performed by sequentially moving instructions and data in and out of memory. The processing capability of a processor is the frequency of the clock and the time required to execute each operation. An FPGA on the other hand is an array of simple logic components which can be "wired" together to perform more complex tasks. An example of this architecture is shown in Figure which shows a small section of the logic in a Xilinx Spartan 6 FPGA. The FPGA is programmed using VHDL (very high-speed descriptive language). VHDL has been compared to assembly from the perspective that the programming is done at a very low level, often dealing directly with the signals. It is however considered a high level language in FPGA development as it is not usually necessary to directly target specific logic elements. During synthesis (equivalent to compiling) the VHDL is converted to RTL (resistor transistor logic) as shown in Figure . This RTL is then mapped onto the FPGA fabric during implementation.

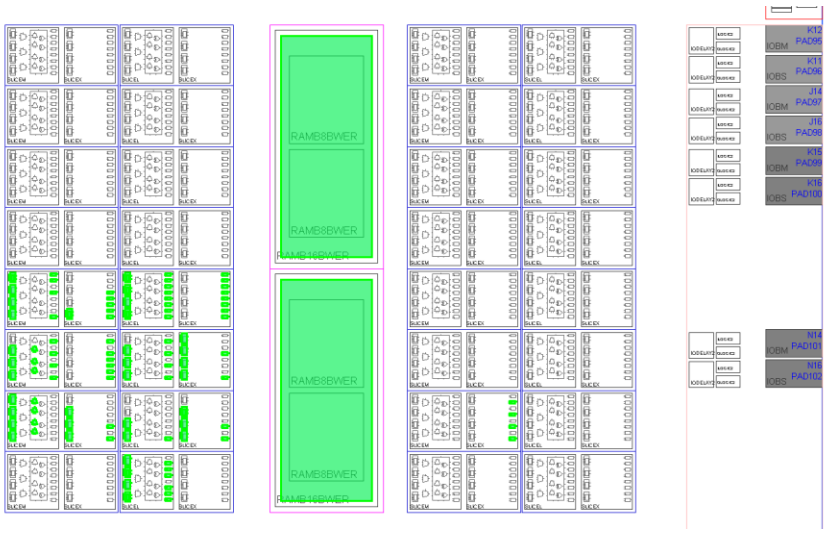

Figure B.1 FPGA Sample Architecture from Xilinx ISE PlanAhead

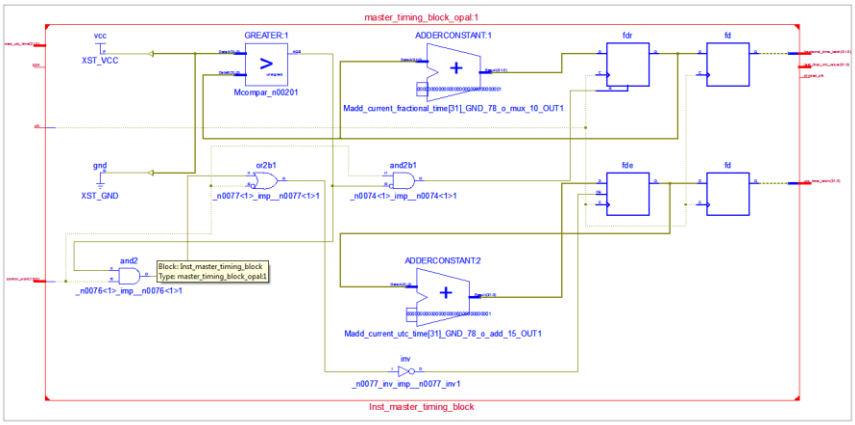

Figure B.2 FPGA RTL Schematic Sample

The benefit of the FPGA in high speed and time sensitive tasks is the distribution of tasks into parallel logic. For example in a system with multiple ADCs, sensors, input and output data streams, an FPGA can perform all the tasks simultaneously without interruption or delay because each task has been mapped to its own logic and runs independent of the others.

In microcontrollers, peripherals like serial interfaces are implemented in built in dedicated logic. As a result the maximum number of peripherals is fixed. Sometimes devices in microcontrollers are multiplexed so that one peripheral or I/O cannot be used simultaneously with another. This limitation does not exist in FPGAs, as long as the logic space and additional I/O ports exist, new peripherals can be added without issue.

With a microcontroller, instructions and data are stored in memory. The size of the memory naturally limits the complexity of the tasks to be performed. An FPGA is similarly limited by its area. More complexity requires more logic. For this reason FPGAs are usually scalable within a family. The VHDL written for a smaller part will run identically on the larger part.