A MULTI-SENSOR PROXIMITY MEASUREMENT SYSTEM ON FPGA FOR AVIONIC APPLICATIONS

ARYAN YAGHOUBIAN

A Thesis
in
The Department
Of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements for the
Degree of Master of Applied Science (Electrical and Computer Engineering)
at

Concordia University
Montréal, Québec, Canada

August 2016

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Aryan Yaghoubian

Entitled: "A Multi-Sensor Proximity Measurement System on FPGA for Avionic Applications"

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

______________________________ Chair
Dr. Dongyu Qiu

______________________________ Examiner, External
Dr.L Wang (BCEE) To the Program

______________________________ Examiner
Dr. H. Rivaz

______________________________ Supervisor
Dr. S. Abdi

Approved by: ______________________________
Dr. W. E. Lynch, Chair
Department of Electrical and Computer Engineering

_ August 2016_

____
Dr. Amir Asif, Dean
Faculty of Engineering and Computer Science

Abstract

A Multi-Sensor Proximity Measurement System on FPGA for Avionic Applications

Aryan Yaghoubian, M.A.Sc. (Electrical and Computer Engineering)
Concordia University, Montreal, QC, Canada, 2016

Inductive proximity sensors are widely used in the aerospace industry for non-contact metal sensing. The main advantages are that they can operate in a harsh environment, providing a rapid response time and a long operational life. To overcome the problem of temperature drift, this thesis presents a novel excitation method that automatically compensates the temperature variations as well as providing precise distance outputs. The inductance and resistance values of the sensor are measured separately. These values can be used to determine the proximity distance and the temperature of the sensor independently. By having this methodology, a Field-Programmable Gate Array (FPGA) design is implemented using look-up tables and DSP (Digital Signal Processing) blocks to accurately measure the proximity distance based on the measured inductance value. Our experimental data shows that we are able to measure the distance in the range of 0 – 5 mm with less than 2% error in the temperature range of -30 °C and +70 °C. Moreover, this method is scaled to support multiple sensors with different internal characteristics up to 10 sensors using a single processing circuit. This is a major improvement over existing electronic circuits which are limited to one sensor type. Finally, an automated test platform is designed to accelerate the test and development process.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Samar Abdi for his continuous support. He taught me in a research environment, there is no supervisor/student hierarchy and what is more important is how to play as a team member and how to think and analyze problems collaboratively. He also taught me how to be precise on every single detail and take everything seriously. Moreover, he taught me how to provide simplicity while tackling complex problems.

Besides my supervisor, I would also like to thank Thales Aerospace, Canada who provided me an opportunity to join their team as an intern, and gave access to laboratory and research facilities. I would like to especially thank Yvon Nazon for sharing his knowledge as well as providing a great detailed supervision during my internship. He taught me no matter how big a task could be, there is always a way to break it down and accomplish it. I also would like to thank Francisco Goncalves, Vincent Jalbert and all the great people that I have this chance to work with at Thales Aerospace, Canada.

I would like to thank my colleagues at Concordia University, Sara Karimi, Ehsan Saboori, Shafigh Psarzad and Umar Aftab for their help and support. I would like to especially thank Faras Dewal who helped me in the project. I also would like to thank Paul Leons who introduced me the project and patiently guided me to the start point.

I would also like to acknowledge MITACS for providing funding to our research. Moreover, I would like to acknowledge CMC for providing research facilities including Xilinx FPGAs. Finally, I would like to thank and dedicate this thesis to my family, whom without their unconditional love it would not be possible to accomplish this work.

To my loving parents,

brother and sisters

Contents

List of Tables

List of Figures

Code Listings

Glossary

ADC	Analog to Digital Converter
ASIC	Application Specific Integrated Circuits
AXI	Advanced eXtensible Interface
CLB	Configurable Logic Block
DAC	Digital to Analog Converter
DSP	Digital Signal Processing
FPGA	Field Programmable Gate Arrays
GUI	Graphical User Interface
IP	Integrated Property
JTAG	Joint Test Action Group
RAM	Random Access Memory
ROM	Read Only Memory
SPI	Serial Peripheral Interface
TCL	Tool Command Language
VHDL	Very High Speed Hardware Description Language
WoW	Weight on Wheels

Chapter 1

Introduction

1.1. Motivation

Recent technological advancements in the aerospace industry are remarkable and aircraft manufacturers always seek to benefit from a variety of wide technologies to boost their performance. For example, Airbus A380-1000 will be equipped with 10,000 sensors in each wing. These sensors provide precise monitoring of different components of the airplane [1].

A common concern often raised in the aerospace industry is related to the aircraft's operational life. The operational life means how many years an aircraft can remain in the service before being scrapped. Various factors can affect the life expectancy of an airplane such as flying conditions, the number of flying cycles per year, maintenance policy of the airline, etc. For example, considering two same airplanes (manufactured exactly the same year), one of them scrapped at 20 years old and one of them is still flying at 36 years old [2]. Broadly speaking, one solution for increasing the life expectancy of an aircraft is investing in aircraft maintenance. Aircraft maintenance means inspecting, repairing and making sure that an aircraft can fly safely during its useful life period. However, the costs associated with maintenance are huge. Globally, in 2014 airlines spent $62.1 billion dollars on Maintenance, Repair and Overhaul, which is 9% of total operational costs [3]. An important contribution of aircraft maintenance is the component maintenance which constitutes 23% of the total maintenance costs [4].

Among these components, sensors play an important role as they provide precise minute to minute monitoring to predict everything from harsh weather conditions to part failures. The data generated by sensors is not only vital for the safety and security of the passengers, but also to reduce the total operational cost of the aircraft. Therefore, providing a reliable and cost-effective sensor solution could potentially save thousands of dollars. The work presented in this thesis aims to reduce the maintenance costs associated with proximity sensors by increasing their reliability, integration flexibility and performance.

1.2. Proximity Sensors

Proximity sensors are widely used in different applications, from automation industry to mobile phones. They can detect the presence of an object without the requirement of physical contact with them, therefore they can also be categorized as non-contact or contactless sensors. Broadly speaking, there are five types of proximity sensors available in the industry for different applications. For example, in some applications, the required detection range could be as high as 100 mm (e.g. automation industry) while in other applications it could be only within 0 to 1 mm range (e.g. mobile industry). We start by describing different types of proximity sensors, then we discuss a particular type that we chose for our avionic application.

- **Inductive proximity sensors:** There is an electromagnetic field around the inductive proximity sensors, when a metallic target enters the sensing zone, this electromagnetic field reduces and by measuring this reduction, the presence of a metallic object can be detected. The advantage of inductive proximity sensors is their ability to withstand harsh environment as well as providing fast responses.
- **Capacitive proximity sensors**: The working principle of capacitive proximity sensors is similar to inductive proximity sensors, except that they generate an electrostatic field instead of the electromagnetic field. Capacitive proximity sensors can detect metallic materials as well as non-metallic materials including liquids, plastics and woods [5]. While capacitive proximity sensors are good for detecting non-metallic materials, they can not effectively withstand harsh environment in comparison to inductive proximity sensors. Furthermore, they operate slower than inductive proximity sensors because the sensing involves charging the capacitive plates [6].
- **Photoelectric proximity sensors**: Photoelectric proximity sensors can measure the distance between the sensor and the target, the presence or absence of an object by transmitting light, often inferred. These sensors are made up of an emitter and a receiver to first, generate the light and then, interpreting it by receiving its reflection. Photoelectric proximity sensors are widely used in automation industries, for example counting the number of bottles in the production line. One drawback of these sensors is that they are more prone to error in dusty and dirty environments as they require more light to perform reliably [7].
- **Ultrasonic proximity sensors**: Ultrasonic sensors generate sonic pulses to detect and object. The distance can be calculated by measuring the total sonic time of flight, which

is a proportion of the distance. These sensors can detect different materials, however as the response signal is not stable for small ranges, it creates a blind zone for objects within that range. Therefore, this property makes them inappropriate for small distance measurements [8].

- **Magnetic proximity sensors**: These sensors are designed for non-contact detection of permanent magnets, they can also detect non-magnetic materials like glasses or wooden wall with the same operating range [9].

Choosing the right sensor depends on the application end, considering various factors such as detection range, the target material, ambient temperature, level of moisture, dust and dirt, maintenance level, etc. In the aerospace industry, inductive proximity sensors are widely used in different parts of an aircraft such as landing gear, cargo loading, doors and flaps. Inductive proximity sensors have interesting properties that make them suitable for this particular industry, including surviving at extreme temperature ranges and withstanding severe conditions. Moreover, they are impervious to dust and dirt with long operational life [10].

1.3. Operating Principle of Inductive Proximity Sensors

The operating principle of inductive proximity sensors is shown in Figure 1.

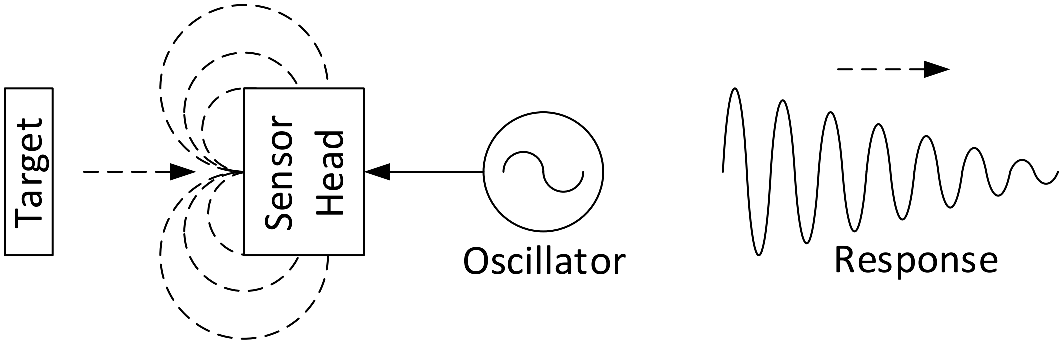


Figure 1: Principle of operation of inductive proximity sensors (from [11])

In [11] Leons *et al.* described the operating principles of inductive proximity sensors as follows:

"An inductive proximity sensor, when excited by an oscillator, generates an electromagnetic field around it. When a metallic target approaches close to the sensor, the target generates eddy current, which reduces the oscillating current in the sensor. The reduction in current is sensed by a circuit that, in turn, acts as a near/far switch. The closer the target, the greater the reduction in current. Therefore, the sensor can be used to determine the distance to target by measuring the amplitude of the oscillating current."

Most inductive proximity sensors used currently in the aerospace industry are switch based, which means by defining a fixed threshold (e.g. 2 mm) they act as a near/far switch. Sometimes a harsh flying condition can affect this threshold by shifting the sensor or the target. While this undesirable shifting might be very small, it is crucial to make sure the near/far threshold is set correctly prior to take-off to avoid any misleading critical information such as falsely indicating the open/close status of the landing gear. Figure 2 shows an example of the near/far behavior of proximity switches.

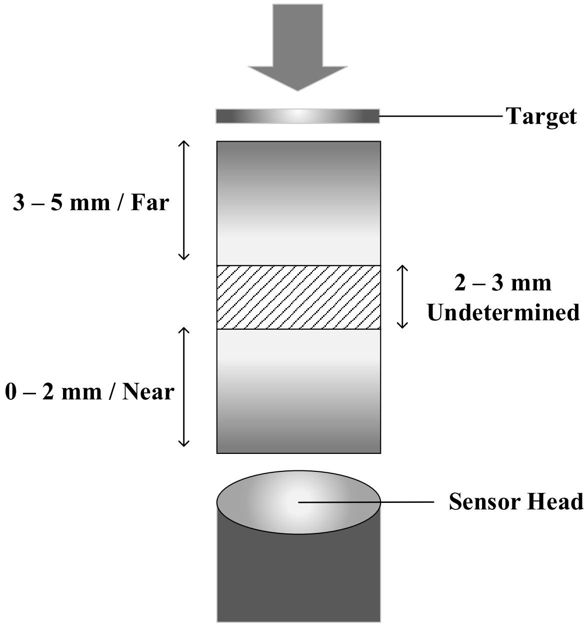


Figure 2: Example of switch based near/far detection

When the target is in the range of 0 and 2 mm the sensor determines the state of the target as near, and in the range of 3 to 5 mm, the sensor determines the state of the target as far. Finally, in the range of 2 and 3 mm it is undetermined, as a safe region is needed to avoid the sudden jump between the near and the far state. For this undermined region, different policies can be applied, one of them is preserving the previous state in the undetermined region until a new state is reached. To explain further, when the target leaves the near zone toward the far zone and enters the undermined zone, the state can be preserved as near until it reaches the far zone which starts at 3 mm. On the other side, when the sensor leaves the far zone toward the near zone and enters the undetermined zone, the state can be preserved as far until it goes below 2 mm.

As in switch based measurement system, there is no reliable way to precisely determine the distance between the sensor and the target, these investigations are done manually by the aircraft maintenance inspectors which is both costly and time-consuming. The way that the manual inspection operates is that after a certain period of time (e.g. 2 months) each inductive proximity sensor needs to be inspected carefully to monitor if any displacements occurred. The sensor/target displacement may happen due to the harsh flight conditions. Therefore, Inspectors have to put special rubber materials between the sensor and the target to accurately measure the displacement. Hence, a precise distance measurement system can replace the existing manual solution to eliminate undesirable maintenance costs. Another drawback associated with fixed based threshold proximity sensors is that they are application based and if a new demand arises in the application they fail to adapt accordingly. Hence, a new measurement system needs to be acquired and integrated into the system that often leads to higher replacement costs. Therefore, being able to adjust the threshold delivers less maintenance cost. Most proximity sensors that currently used in aerospace industry have the nominal range of 2-3 mm with the resolution of 0.1 mm [12, 13, 14]. Increasing the sensor's nominal range and its resolution can increase the measurement accuracy as well as reliability that leads to a lower maintenance cost. The work done in this thesis aims to precisely quantify and measure the distance between the sensor and the target.

1.4. Classification of Inductive Proximity Measurement System

Broadly speaking, inductive proximity sensors can be categorized into two types: active and passive. Active sensors have their own excitation circuitry and distance measurement system. Active sensors are able to output a DC voltage corresponding to the distance between the sensor and the target. On the contrary, passive sensors need to be excited from an external source using a cable. Therefore, the electronic circuits along with processing methodology need to be designed for passive sensors to externally excite them. A typical cost for active sensors used in avionic applications is in the range of $3000 while the cost associated with passive sensors are about a third or less.

1.5. Design Goals and Challenges

The inductive proximity sensors offered by different manufacturers are strictly guarded intellectual property. Therefore, understanding the underlying behavior of the sensor and finding a way to derive its parameters is challenging. Moreover, each sensor offered by a different manufacturer might have different characteristics, such as various internal resistances. As a result, understanding each sensor model to build a generic proximity measurement system that supports different sensor models is vital in our design. Moreover, the electronic circuits offered by a manufacturer only operates with their own proximity sensors. Therefore, supporting various proximity sensors from different manufacturers is not only costly but also requires more effort in the integration part. One of our challenges is, how to integrate inductive proximity sensors with different characteristics in one system.

Another challenge is overcoming and compensating the temperature drift. Temperature can vary a lot during a flight which can change the behavior of the sensor. Hence, making the measurement system robust to such temperature drift is crucial to maintaining a reliable operation accuracy. Our design goal is achieving a measurement system that minimizes undesirable consequences hat occurs due to the temperature drift.

The measurement range for inductive proximity sensors is 2-3 mm. This is sufficient for normal operation, however, there is an interest from aerospace industry to increase this range for reliability purpose. Therefore, our challenge is increasing this range to 5.00 mm with less than 2% error. For example, at 2 mm we should not see measurement errors more than ±0.04 mm. This makes the measurement system more robust to undesirable external changes and addresses the reliability concerns raised by the aerospace industry. However, above 3.00 mm sensor's sensitivity to the target decreases significantly, therefore, makes it harder to accurately measure the distance in 3-5 mm range.

An important sub-system of an aircraft is the Weight-On-Wheels (WoW) system. WoW provides vital information such as whether the aircraft is on the ground or in the air. Proximity sensors used in landing gear determines the WoW status. This is an important info required by different systems of aircraft and if the WoW system malfunctions, it can lead to major system failures. WoW has a real-time constraint of 400 ms, including the system start-up time and distance measurement. Therefore, it is vital to meet this timing requirement by decreasing the proximity measurement time. The real-time constraint of our proximity measurement system is 25 ms which is a refresh rate of 400 Hz. The summary of design requirements is listed in Table 1.

Table 1: Proximity Measurement System Design requirements

Parameter	**Requirement**
Distance range	0 mm – 5 mm
Distance percentage error	Less than 2%
Measurement refreshes rate	40 Hz
Temperature	-30 ^{O}C to +70 ^{O}C.
Number of Sensors	Up to 10 sensors
Sensor Type	Sensors with different internal parameters

1.6. Thesis Contribution

The main contributions done in this thesis are presented as follows:

1. **Current Ramp Excitation Method.** The theory behind current ramp excitation method is generating a response signal that provides a new computational method to reduce the temperature drift. The main advantage of the current ramp excitation method is that it automatically compensates the temperature variation.
2. **Supporting Multiple Inductive Proximity Sensors with Different Characteristics.** Existing inductive proximity sensors do not necessarily have the same internal characteristics. For example, one sensor have an inductance range of 4-6 mH while another sensor has an inductance range of 20-50 mH. Therefore, proper compensation methods need to be implemented to support different inductive proximity sensors. The advantage of our computation method is that it is capable of supporting inductive proximity sensors with different characteristics. Moreover, up to 10 sensors can operate using a single processing circuit.
3. **Test and Development Automation.** Once the system is implemented, a large amount of time is devoted to testing. A variety of tests need to be run to make sure the requirements are met correctly and the measurements are repeatable. As we seek to meet the requirements from different vendors with different test parameters, it can be tedious if the proper test procedures are not chosen. Therefore designing an automated test platform to accelerate the test and development procedure can significantly decrease the development cost. This test platform not only eliminates the human errors

but also makes the measurements more repeatable and robust which can save time and reduce the overall design cost.

1.7. Thesis Organization

The remainder of this thesis is organized as follows. In chapter 2, we discuss the background and related work that have been done for inductive proximity sensors. Then, we briefly explain our main processing platform which is an FPGA (Field Programmable Gate Array) and we introduce its special resources. Moreover, we explore the voltage step excitation method on a well known off-the-shelf inductive proximity sensor offered by Crouzet [12], focusing how to compensate the temperature variation problem. In chapter 3, we describe the current ramp excitation method, explaining how it addresses the temperature drift problem efficiently. We also describe our hardware implementation of current ramp method and observe the experimental results at room temperature and extreme temperature ranges. In chapter 4, we start by analyzing the characteristics of different sensors and we move toward proposing a new method capable of supporting up to 10 sensors with different internal characteristics. Moreover, we compare different inductive proximity sensor types, both at room temperature and at extreme temperature ranges. In chapter 5, we explore how to automate the test and development procedures and we discuss its advantages and benefits. Finally, conclusion and suggested future works are described in chapter 6.

Chapter 2

Background

2.1. Related Work

A variety of inductive proximity designs targeted for different applications has been proposed. In [11], *Leons et al.* proposed a pulse excitation method that integrates a region in the response signal leads to quantitative outputs. The accuracy of this measurement system is 89% with the resolution of 0.1 for 0-5 mm distance range at room temperature. However, this method did not address the temperature drift problem. In [15], *Guo et al.* presented an analog-digital mixed measurement method based on 2D look-up tables capable of producing quantitative outputs at room temperature as well as extreme temperatures. This method also used pulse excitation method and the key idea behind it is to separate the sensor's resistance component and inductance component to find the sensor to target distance, based on the inductance component. The guaranteed actuation distance of this method is 4.50 ± 0.20 mm which is below 4% in the temperature range of 20 °C to 110 °C. However, no experiments have been suggested for extreme temperature ranges below 20 °C. Moreover, this method did not address how to support multiple sensors with different internal characteristics. A combined inductive-capacitive proximity measurement system is suggested in [16]. While this method can detect the presence of an object, it does not provide any quantitative outputs and as a result, cannot accurately measure the distance. A variety of analog designs has been suggested using inductive proximity sensors. Analog approaches are popular because they are easier to implement, however, they only provide analog outputs and they are not capable of accurately measuring the sensor to target distance. Switch based analog measurement system involves exciting the sensor using a pulse excitation and comparing the discharge waveform using a comparator to evaluate the inductance value and detect if the target is close to the sensor or not. *Nabavi et al.* proposed a measurement method in [17] for the excitation frequency above 22 MHz. While this method compensates the temperature drift, it cannot produce quantitative outputs. Moreover, the distance range specified in this method is limited to 1.5 mm.

2.2. FPGA

FPGAs are widely used in different applications, from medical image processing to the aerospace industry. FPGAs are integrated circuits comprise of a matrix of configuration logic blocks (CLBs) which are connected via programmable interconnects. What makes FPGAs popular is their ability to be reconfigured as many times as a designer want. In other words, FPGAs can be reprogrammed to deliver a flexible functionality for an application while ASICs (Application Specific Integrated Circuits) are designed to serve specific design tasks. Once ASICs are manufactured they cannot be reconfigured. Moreover, the hardware prototyping of FPGAs is much faster than ASICs as they are already built and mass-produced by FPGA manufacturers. Most FPGAs have special resources and features that can accelerate the design. We explore some of these resources in the two following sub-sections.

2.2.1. Memory Resources

Most of the digital designs need internal memory to store local data such as coefficients and look-up tables. Often these data are not grouped in the same size and shape, therefore flexible memory allocation is required for higher performance. For example, if we allocate a large memory to store small data, not only we waste memory, but also we produce undesirable power. Two most common memory types used in FPGAs are BRAMs (Block RAMs) and distributed RAMs. Considering one commercial FPGA example, the BRAM can store up to 36 Kb of data and can be configured either as one 36 Kb RAM, or two independent 18 Kb RAMs. Figure 3 shows the block diagram for a BRAM.

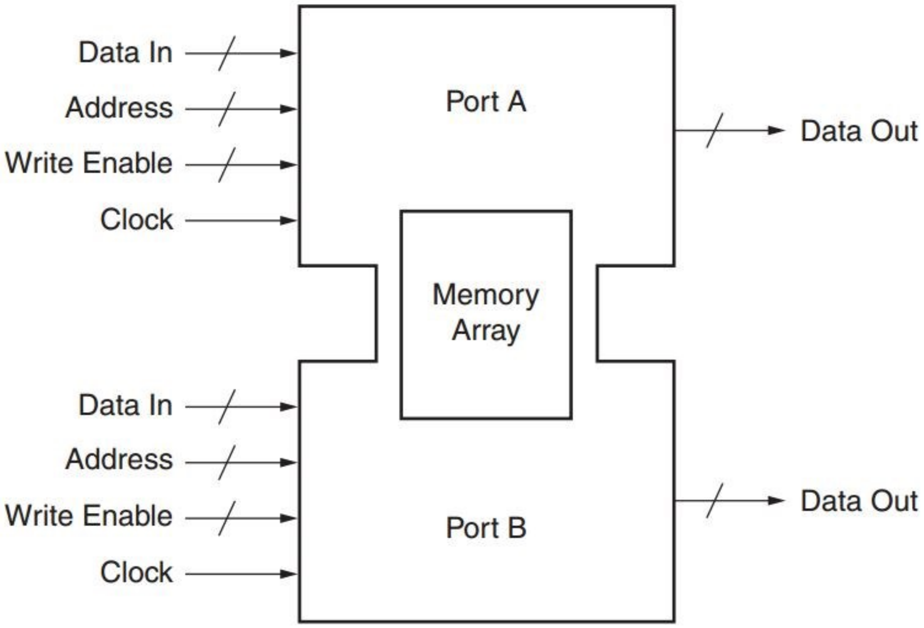


Figure 3: BRAM block diagram (from [18])

Each BRAM is configurable by specifying the data width and address depth. For example, an 18 Kb BRAM can be configured as a $16K \times 1\ bit$ RAM with an address size of 14 bits (Address[13:0]) and data width of 1 bit or as a 512×36 RAM or any other combination of data width and address depth respecting the total memory size. Configuration of the BRAM totally depends on the application. BRAMs can also be used as ROM memory (Read Only Memory). In this mode, there is no write operation involved and the pre-calculated data are stored for read operation only. This feature along with the flexibility of memory size, make BRAMs suitable for implementing look-up tables.

The second type of memory storage is distributed RAMs. FPGAs consist of an array of Configuration Logic Blocks (CLBs) and each CLB consists of smaller elements, among these elements, 6-input LUTs can be configured as a $64 \times$ 1-bit RAM or two $32 \times$ 2-bit RAMs. These synchronous RAM resources are called distributed RAM. Other configurations are also possible [18]. The small memory size of distributed RAMs makes them suitable for smaller memory allocation such as coefficients while leaving the BRAM resources for larger memory allocations that might be required by other resource consuming tasks.

2.2.2. DSP Resources

DSP (Digital Signal Processing) slices are special resources designed to implement arithmetic functions such as multiplications and accumulations benefiting high performance, low power combination. Understanding the underlying architecture of these dedicated DSP slices is vital for design optimization. Moreover, these DSP blocks are not limited to signal processing applications, they can also be used in a verity of different applications, such as pattern detection, wide dynamic bus shifters, memory address generator, etc. [19]. Figure 4 shows the basic DSP48E1 slice which used in Xilinx's FPGAs.

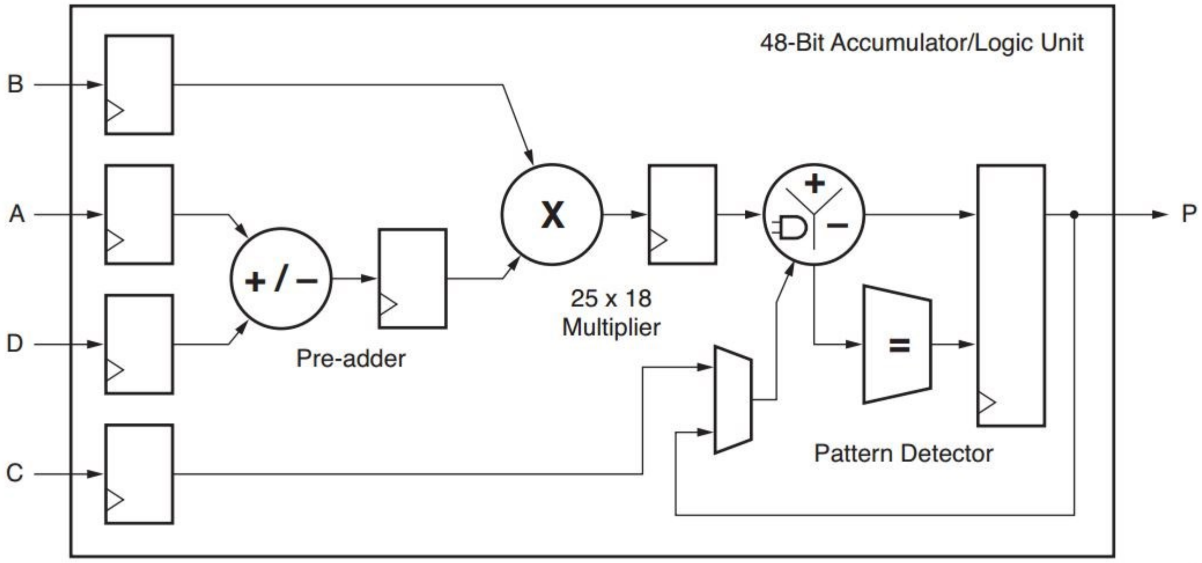


Figure 4: Basic DSP48E1 slice Functionality (From [19])

DSP blocks can also be cascaded to form larger arithmetic functions. For example, the P output of one DSP48E1 slice can be connected to another PCIN (P carry-in) input of an adjacent DSP48E1 slice for the next level of computation. In our design, we exploit DSP48E1 features to enhance our performance while meeting our low-power design requirements.

2.3. Inductive Proximity Sensor Modeling

The passive sensor produces a response signal when excited by an external source. This signal can be analyzed to derive the sensor model. Moreover, different excitation methods can be applied for further analysis in order to find a reliable method to calculate the distance. Figure 5 shows the response signal of a Crouzet passive sensor using a simple voltage step excitation. We used a commercial off-the-shelf inductive proximity sensor offered by Crouzet [12].

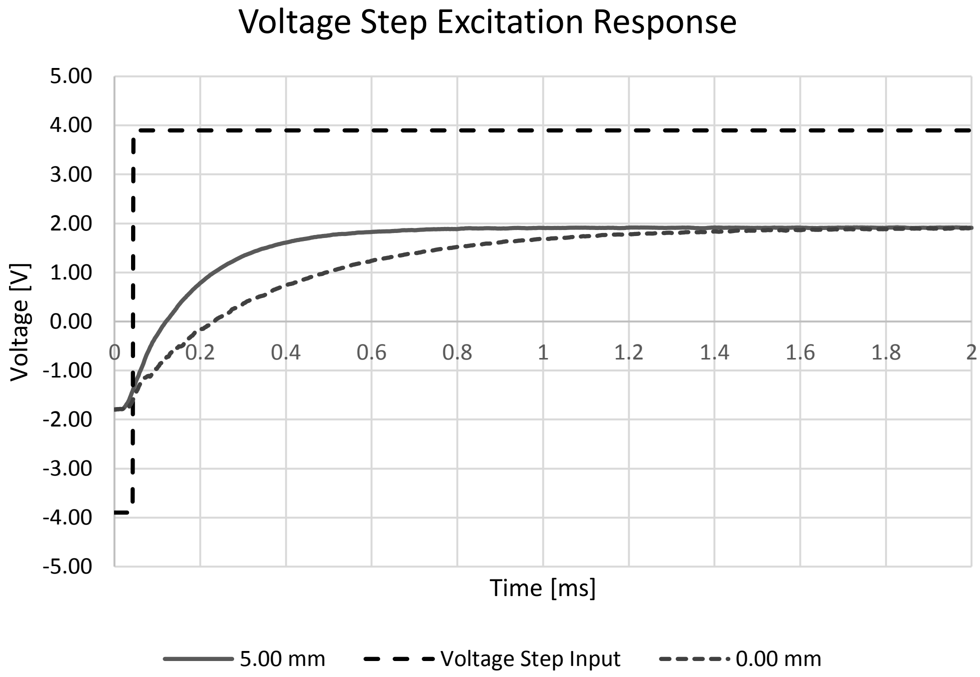


Figure 5: Step response of a Crouzet passive inductive proximity sensor

Excitation responses are compared to each other at the minimum and maximum distances between the sensor and the target which are 0.00 mm and 5.00 mm respectively. This comparison shows that the more we increase the distance, the more the sensor impedance drops which leads to a higher slope for the response signal. Furthermore, based on the experimental data, at all distances, the current through the sensor settles to the same value. Considering this observation, a first-order RL series can model the sensor. Figure 6 demonstrates a simplified version of the inductive proximity sensor.

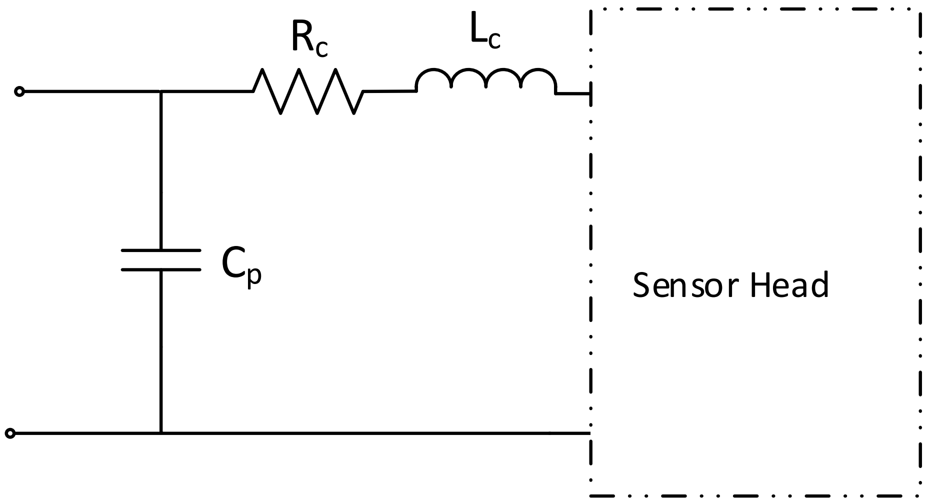


Figure 6: RL Model of the proximity inductive sensor

L_i represents the inductance of the sensor and R_i shows the resistance of the sensor. L_c is the equivalent inductance of the cable and R_c is the equivalent resistance of the cable and the cable length used in all our experiments is 100 mm unless specified explicitly. C_p is the equivalent parasitical capacitance. In the remainder of this thesis, we refer to the total value of inductance by L which is equal to$L_i + L_c$ and the total resistance value by R which is equal to $Ri + Rc$. Moreover, experimental data showed that the effect of the capacitance is negligible thus it can be ignored.

By understanding the sensor model different excitation methods can be applied to measure the distance based on the changes in the response signal. Leons *et al.* investigated sinusoidal excitation method in [11] and by observing no significant changes in the response signal beyond 2.5 mm found that it is not an ideal acquisition method. In the next section, we investigate the voltage step method and possible solutions to minimize the temperature drift effect.

2.4. Voltage Step Excitation at Room Temperature

Voltage step excitation is one of the simplest approaches to see how the sensor behaves when the distance changes. In Voltage step excitation, the current response can be derived in Equation 1 where $\tau = \frac{L}{R}$ is the time constant.

$$i(t) = \frac{V_{DC}}{R} + \left(i(0) - \frac{V_{DC}}{R}\right) \exp(-\mathrm{t}/\tau)$$

Equation 1: current response equation of the circuit

Experimental results showed that the inductance L, changes significantly with distance, however, slightly affected by temperature variation. On the other hand, resistance R, changes significantly by temperature, however, slightly affected by distance variation. Hence, we can deduce that the inductance L, is the distance dependent parameter of the sensor. As a result, extracting L from the response signal can ideally be used to calculate the distance. However, this computation is extremely complex as it requires an exponential curve fitting in real-time. Therefore, considering our real-time constraint it is infeasible to implement this approach on an embedded platform.

An alternative solution is to identify the response region that has a maximum correlation with L. Experimental results at different distances showed the most variation of L occurs at the beginning of the response signal. This region is shown in Figure 7 and we refer to it as the region of interest.

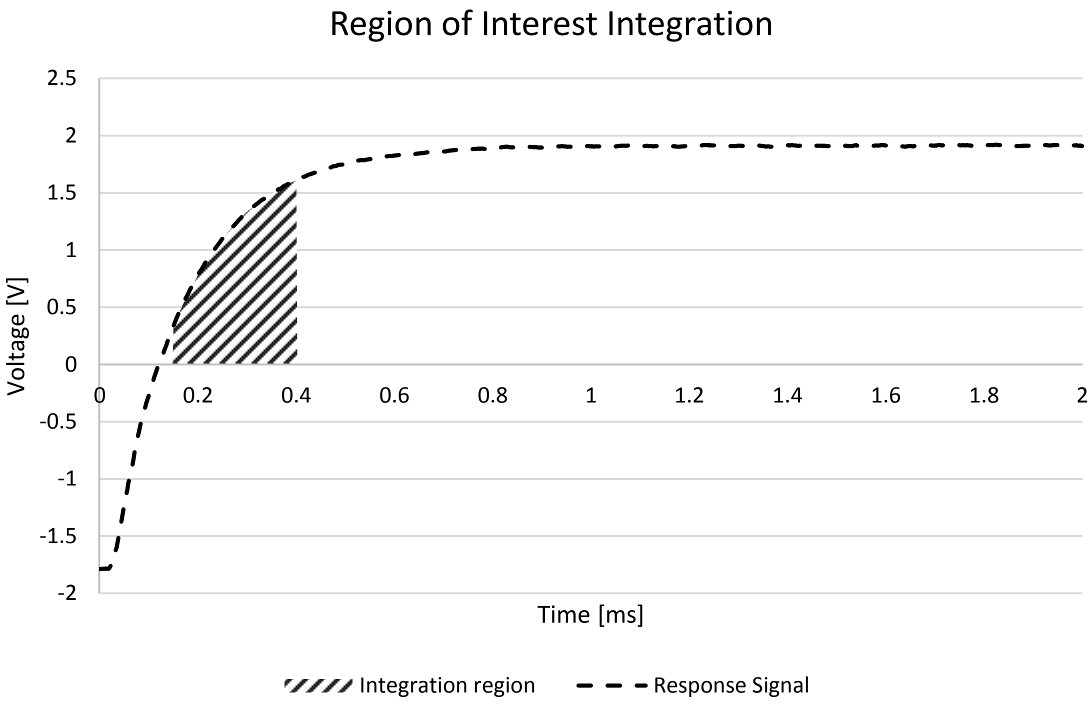


Figure 7: Region of interest over the response signal

Therefore, a feasible computation method is to integrate the response signal in this region as a distance proxy. Moreover, it does not have the computational complexity of curve fitting. When characterizing the sensor, the integral value is measured for each distance in the range of 0 – 5 mm at 0.1 mm and stored in a look-up table. During distance measurement computation, the integral in the region of interest is computed in real-time and then used to look up the distance using an integral/distance look-up table. This approach enables us to compute the distance without extracting the exact internal parameters of the sensor. Leons *et al.* investigated this excitation method in [11] and found a reliable accuracy of 89% at room temperature, given the distance range of 0 – 5 mm with 0.1 mm resolution using the Crouzet sensor.

2.5. Voltage Step Excitation at Different Temperature

Although the region of interest is mostly dependent on L (distance) it is not totally independent of R and therefore, a slight change in temperature can result in a different integral value in the region of interest and therefore, to a different proxy of the distance. For example, considering the variation of the integral value at a fixed distance of 4.0 mm over a temperature range of -30 ^{O}C and +70 ^{O}C, we can observe more than 25% variation in the integral value. This variation is shown in Figure 8. Experimental data showed that this amount of variation in the integral value can result in a ±0.5 mm error in the measurement which is not acceptable.

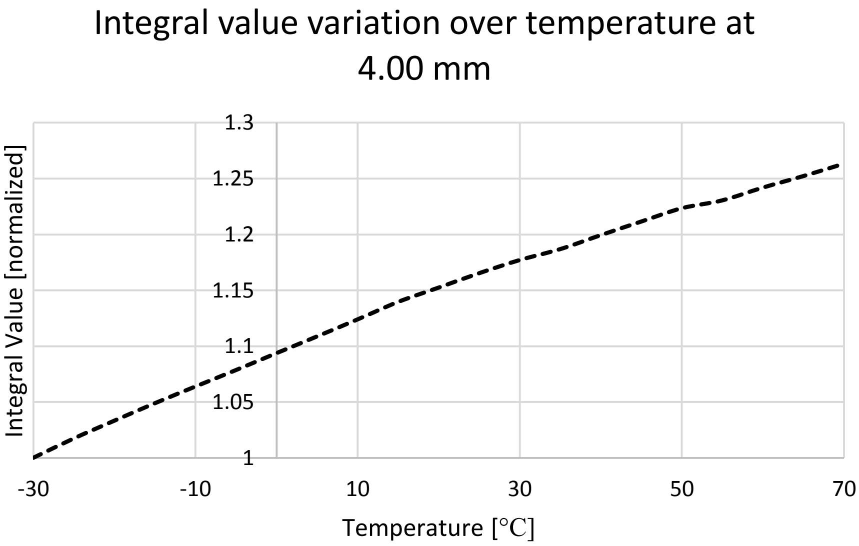


Figure 8: Integral variation over temperature at 4.00 mm

One approach to compensate the temperature drift is finding a correlation between the temperature and its corresponding integral value in the region of interest. This can be done by integrating the saturated region of the response signal which is shown in Figure 9 as a proxy for temperature. This integration correlates with R, the temperature dependent component of the sensor, thus, it can be used to associate each region of interest's integral value to its temperature proxy.

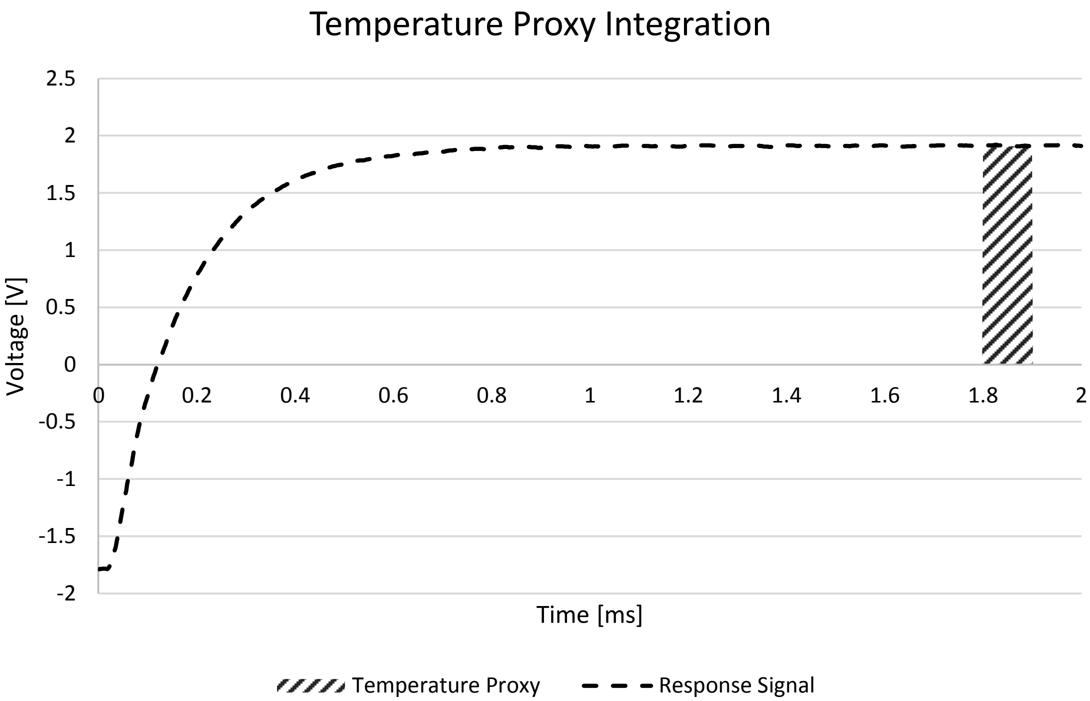


Figure 9: Integrating the saturated region as a temperature proxy

After gathering the distance integral value and temperature integral value at all distances in the range of 0.0 mm and 5.0 mm with 0.1 mm resolution in the temperature range of -30 ^{O}C and +70 ^{O}C, we can plot this correlation in a 3D surface using MATLAB which is shown in Figure 10. Temperature proxy is denoted by A_T and distance proxy is denoted by Sigma. This graph shows that the temperature affects the distance proxy and as a result, leads to a different distance.

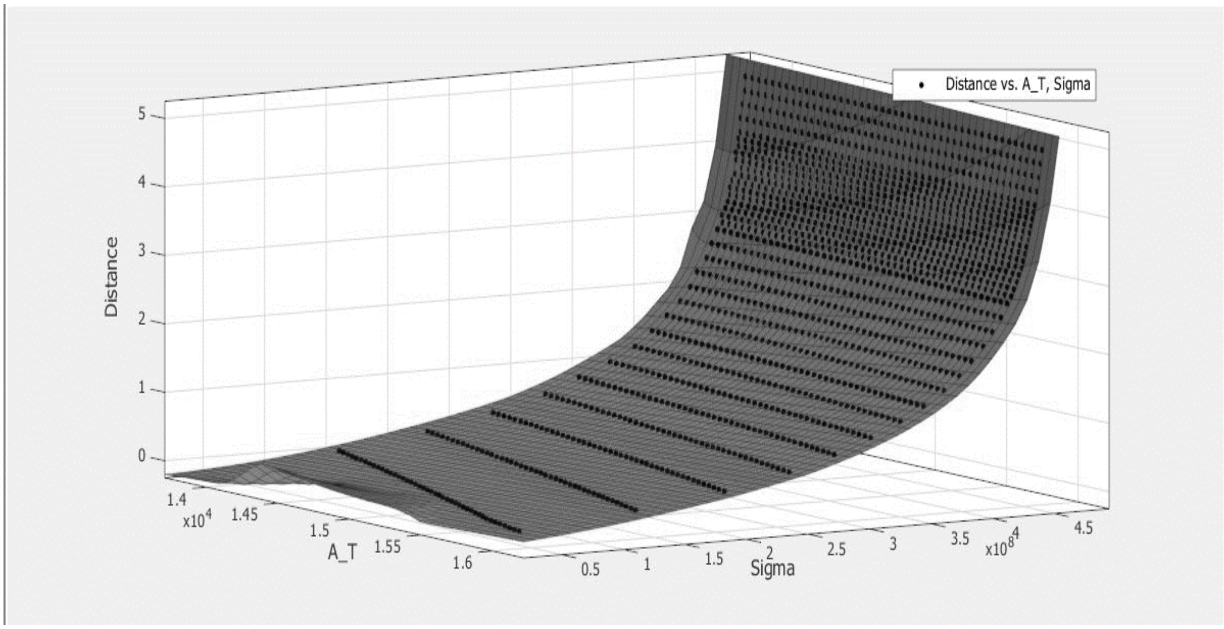


Figure 10: Distance Proxy and Temperature Proxy correlation

Considering the 3D plot shown in Figure 10 a 2D polynomial curve can be used as a curve fitting method to compute the distance. However, finding a proper 2D curve that estimates the distance accurately needs a high order polynomial function which is extensively computationally complex, thus implementing this approach is not suitable for our application considering our real-time constraint.

Another approach is using 2D look-up tables to store all distance look-up tables per temperature proxy values. In this approach, the size of the look-up tables is dependent on temperature granularities. For example, a calibration with 5 ^{O}C resolution in the range of -30 ^{O}C to +70 ^{O}C generates 21 distance look-up tables. Experimental data showed that using this approach leads to an accuracy level of 0.2 mm which is still not promising. The reason is that even 5 ^{O}C step for look-up tables is not small enough to overcome the sensitivity of the sensor to the temperature. Hence, smaller temperature steps are needed which leads to a higher memory utilization. Given our multiple sensory type requirement in our proximity measurement system, it is not feasible to fit such large look-up tables in our embedded platform. Moreover, another obstacle associated with both approaches is that the calibration process requires a huge amount of time devoted to gathering the integral values over different temperatures which are not only time-consuming but also costly as it requires special equipment. Based on this observation, although the voltage step method shows promising results at room temperature, it fails to properly address the temperature drift issue.

2.6. Summary

In this chapter, we discussed the background and related work that is done in inductive proximity measurements. We also briefly described FPGA and explored its special resources that can be used for design optimization. Moreover, we modeled an inductive proximity sensor using a first order RL series and concluded that this modeling helps us to understand the underlying components of the sensor. Furthermore, the voltage step excitation method has been analyzed both at room temperature and at different temperatures, its drawbacks have been explained, such as large computational overhead to overcome the temperature drift. In the next chapter, we propose a new method to efficiently and effectively address the temperature drift problem.

Chapter 3

Current Ramp Excitation Method

3.1. Current Ramp Excitation

As we have seen in chapter 2, the main drawback with voltage step method is that it was unable to distinctly specify the resistance and inductance components of the proximity sensor. Therefore, it could not effectively measure the distance based on the inductance value. A new approach would be suggesting an excitation method that automatically separates the inductance, L, and the resistance R, hopefully leading to a lower computational overhead. The voltage response of the sensor circuit can be derived as follows:

$$V(t) = R \times i(t) + L\frac{di}{dt}$$

Equation 2: voltage-response equation of current ramp

The L contribution of the response signal in Equation 2, is based on the rate of change in current. That being said, exciting the sensor with a fixed rate of change in current can potentially lead to a computational method capable of separately specifying the L and R contributions. This excitation signal is shown in Figure 11 and is called current ramp excitation method which the current input is simply a smoothed triangle function. Experimental data showed that smoothing the excitation signal results to a more stable and less noisy response, therefore results to a higher performance thus, we used a smoothed version of the triangle signal for our current ramp excitation method.

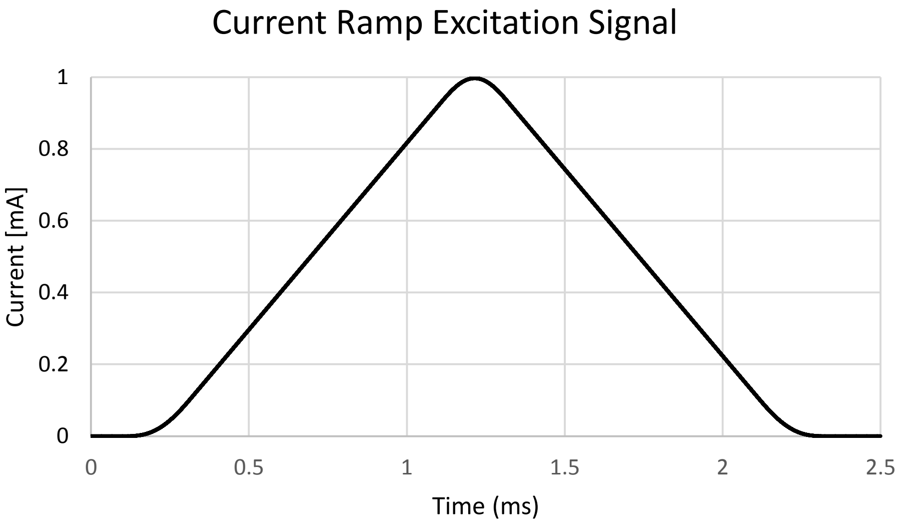


Figure 11: Current ramp excitation signal

3.2. Single Sigma Distance Measurement Method

By applying the same principle that we discussed in section 2.4for the voltage step method, we can integrate over the region of interest of the current ramp excitation response to see how accurate it can measure the distance. Figure 12 shows a response signal using the current ramp excitation signal and the new integration region is indicated by sigma.

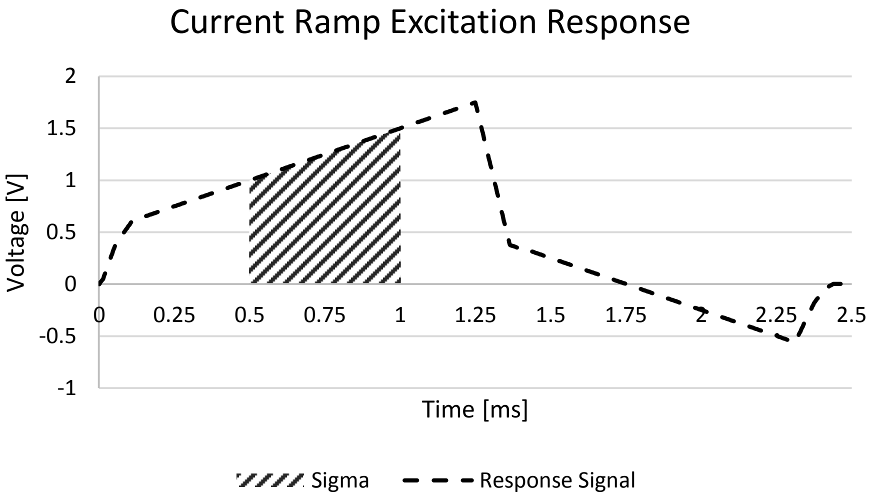


Figure 12: Current ramp response signal

Using single sigma method showed a reliable accuracy below 2.5 mm at room temperature. However, very poor results have been observed above 3.0 mm. The distance measurement error at room temperature was more than 0.2 mm in the range of 3-5 mm. These errors have been shown in Figure 13 for the range of 2.5 -5 mm with 0.1 mm resolution.

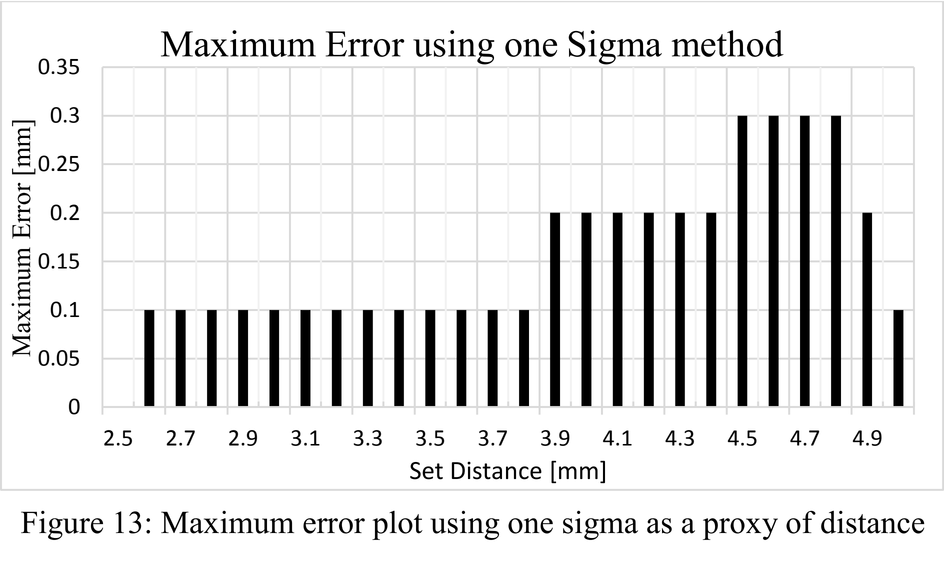


Figure 13: Maximum error plot using one sigma as a proxy of distance

Moreover, more than 25% variation is observed for sigma values over the temperature experiment. This large variation in sigma can result in a large measurement error. The sigma variation for temperature range of -30 ^{O}C and +70 ^{O}C at 4.0 mm is shown in Figure 14. This is expected, as in this sigma region, the R contribution is still involved and therefore, this method fails to meet the accuracy requirement at different temperatures.

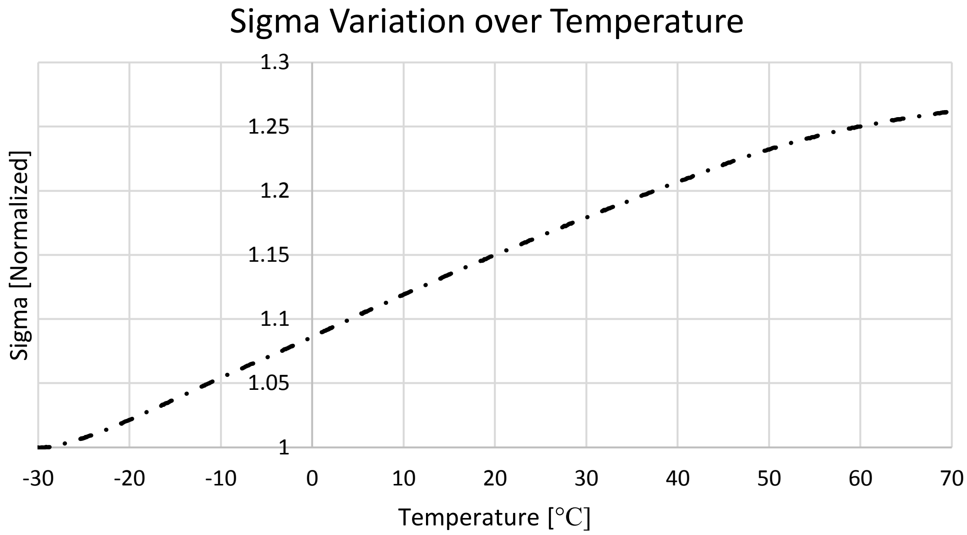


Figure 14: Sigma variation at 4.00 mm over -30 ^{O}C to + 70 ^{O}C temperature range

3.3. Double Sigma Distance Measurement Method

Considering the accuracy results presented in section 2.4 and section 3.2 for voltage step method and single sigma method indicates that as long as the R component is involved in the distance proxy, it is hard to achieve a reliable operational performance at different temperatures with a low computational complexity. Therefore, being able to exclude the resistance contribution from the inductance contribution can effectively lead to a computational method that reduces the temperature drift. Double sigma method aims to provide such computational advantage by separating the L component and R component of the sensor. The excitation response signal for a current ramp input with 2 ms period is shown in Figure 15. In this response, two sigma regions are indicated by Σ_1 and Σ_2. Both Σ_1 and Σ_2 have R and L contributions. However, as di/dt is a constant value in the stable region of the response signal, therefore, by introducing a proper scaling factor α, and using the combination of Σ_1 and $\Sigma_{2,}$ R contribution can be eliminated to derive a distance proxy value, $\Sigma_{D,}$ which only contains the L contribution and as a result purely depends on the distance. On the other hand, by introducing another scaling factor β, the L contribution can be excluded to derive a temperature proxy value, $\Sigma_{T,}$ which ideally only depends on the temperature of the sensor.

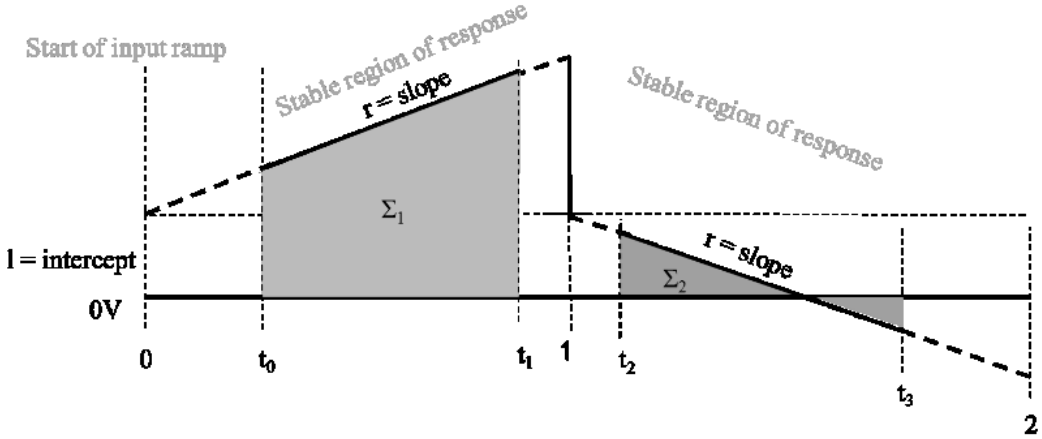


Figure 15: Illustration of Σ1 and Σ2 in current ramp

Σ variables, time stamps, *l* as intercept and *r* as slope are shown in Figure 15. It should be noted that *l* is distance dependent and *r* is temperature dependent. The detailed calculation of deriving Σ_D and Σ_T have been shown below in Equation 3.

$$\alpha = \frac{[(2 - t_2)^2 - (2 - t_3)^2]}{(t_1^2 - t_0^2)}$$

$$\beta = \frac{(t_3 - t_2)}{(t_1 - t_0)}$$

$$\sum{}_1 = l \times (t_1 - t_0) + \frac{1}{2} \times r \times (t_1^2 - t_0^2)$$

$$\sum{}_2 = l \times (t_3 - 2) + \frac{1}{2} \times r \times [(2 - t_2)^2 - (2 - t_3)^2]$$

$$\sum{}_D = \sum{}_2 - \alpha \times \sum{}_1$$

$$\sum{}_T = \sum{}_2 - \beta \times {}_1$$

Equation 3: $\sum_D$ and $\sum_T$ Derviation

Therefore, a viable computation method is to integrate the response in Σ_1 and Σ_2 regions of interest and deriving Σ_D and Σ_T. The Σ_D value serves as a proxy for distance and the Σ_T value serves as a proxy for temperature. Furthermore, by introducing proper scaling factors, the precise value of inductance L and resistance R can be derived from Σ_D and Σ_T, respectively. During the sensor characterization phase, the inductance for each distance is measured and stored to generate the inductance/distance look-up tables. During deployment, the inductance value is measured in real-time and the corresponding distance is calculated using look-up tables and DSP Blocks. As such, this empirical method enables us to characterize a sensor and create a distance-computation logic based on the L parameter of the sensor that automatically addresses the temperature drift. Moreover, this method has significantly less computational effort in comparison to the previous methods. Figure 16 shows Σ_D variation over temperature and Figure 17 shows Σ_T variation over temperature. The values have been normalized for a better illustration.

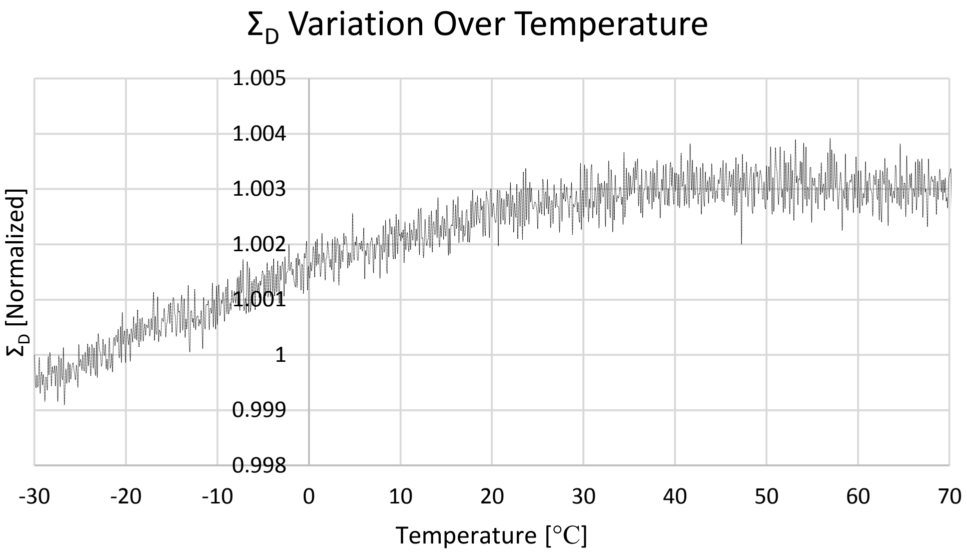


Figure 16: ΣD Variation over temperature at 4.00 mm

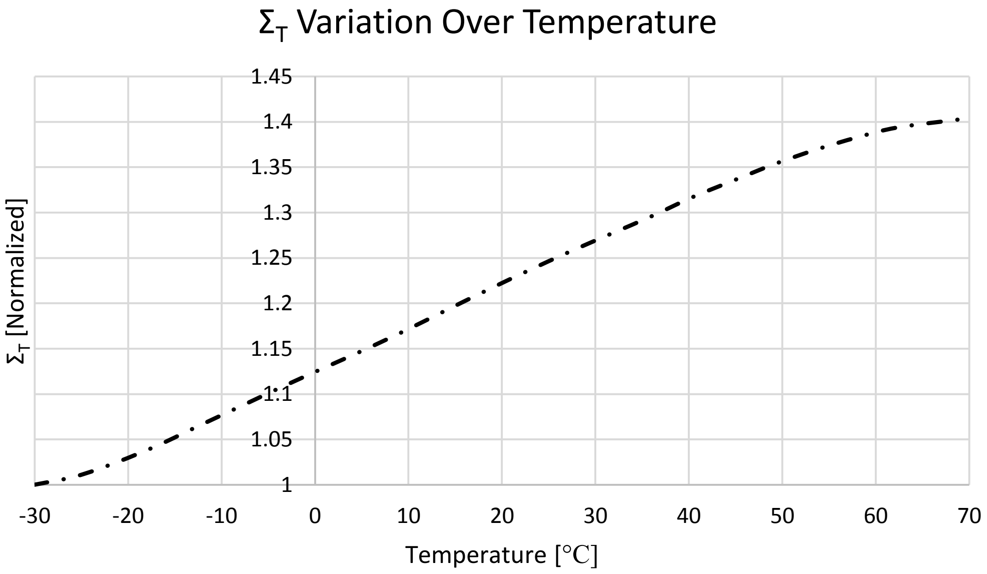


Figure 17: ΣT Variation over temperature at 4.00 mm

As shown in Figure 16, the Σ_D variation is less than 0.4% which is a major improvement in comparison to what we had in the previous methods. For example, it was more than 25% in the single sigma method. This property significantly reduces the temperature drift by using Σ_D as a proxy for distance. On the temperature side, Figure 17 shows 40% variation for Σ_T. This is expected and proves that Σ_T is mostly dependent on R component and therefore, temperature. In the next section, we will explore the hardware implementation of the current ramp method and then we will discuss the experimental results based on this new approach both at room temperature and different temperatures.

3.3.1. System Overview

The abstract illustration of the hardware platform is shown in Figure 18. This platform consists of the following components: FPGA, DAC, ADC, clock Generator, a proximity sensor and a motorized linear stage. The chosen proximity sensor is FT84798 from Crouzet [12] which is a commercial off-the-shelf sensor used in avionic applications.

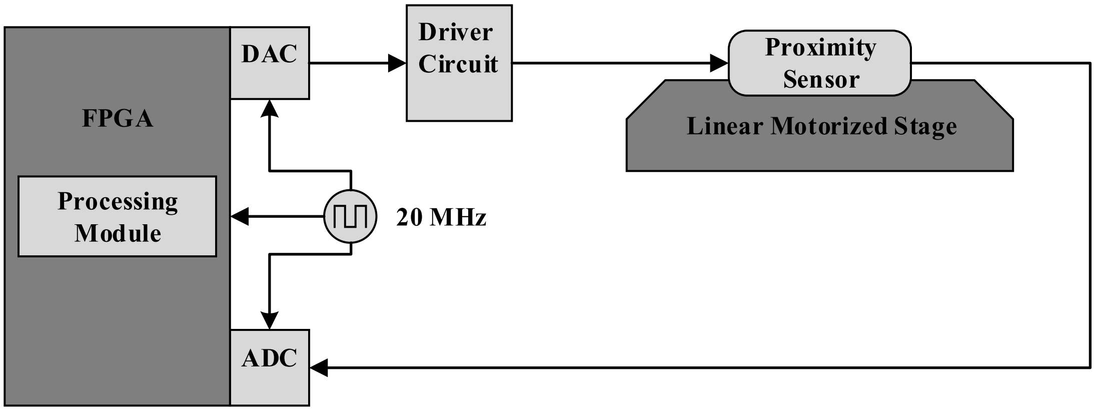


Figure 18: Current Ramp Implementation System Overview

The clock generator is used to distribute the clock throughout the various components in the system and sync them along at the clocking rate of 20 MHz. FPGA is the main processing core of our system and is responsible for a variety of different computational tasks, such as generating the current ramp excitation pattern as well as performing required computations to calculate the distance.

DAC (Digital to Analog Converter) receives the digital excitation pattern from the FPGA and throughout the driver circuit provides the proper analog output to excite the proximity sensor. The driver circuit which basically consists of amplifiers and resistances creates the necessary driving voltage to properly drive the sensor. ADC (Analog to Digital Converter) on the other hand, is used to read back the response signal from the proximity sensor and converts it to digital format with the same global clock rate which is 20 MHz. After sampling the output, ADC forwards the signal to the FPGA to be used by the processing core for real-time computations. The hardware implementation is done using Xilinx Vivado Design Suite software and is coded in VHDL along with IP Integrator tool from Vivado Design Suite.

The linear stage is acquired to accurately position the distance between the sensor and the target. It is also used to automate the calibration and test procedures which are explained in details in chapter 5. In the following subsections, we will describe our processing modules and explain the algorithm used to calculate the inductance and the resistance for calibration purposes. Moreover, we will describe how we compute the distance in real-time.

3.3.2. Processing Module

The mathematical theory behind the double sigma approach described in section 3.3 is the main idea for our processing core. By exploiting dedicated resources such as BRAMs and DSP Blocks, processing core implements this methodology. The abstract model of the processing core is shown in Figure 19.

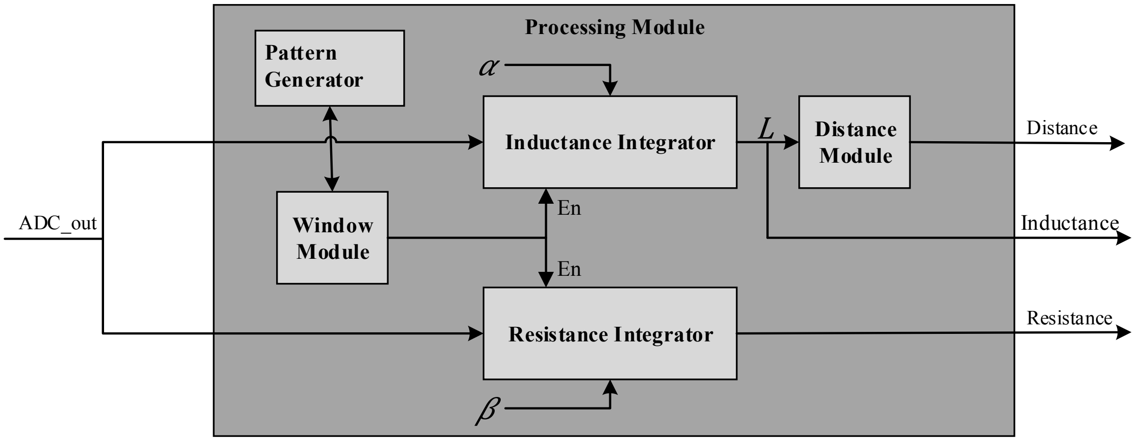


Figure 19: Processing Module Abstract Model

Processing Module receives the digital output from the ADC and forwards it to the integrator blocks. The two integrator blocks, i.e. Inductance and resistance integrators are implemented using DSP blocks and they perform the integration over the response signal for a predefined integration. Window module defines this region by having the flexibility to assert the enable signals. Moreover, having the window module makes the design more flexible as it allows the integration blocks to modify their integration periods by changing a reference look-up table associated with that.

The total period of the response signal is 2.5 ms and the integration period that we used for Σ_1 and Σ_2 (shown in Figure 15) is 400 µs for each, which makes the total integration time of 800 µs. Σ_1 starts the integration at 690 µs and ends at 1090 µs. Σ_2 starts at 1690 µs and ends at 2090 µs. The reason that there is a delay at the beginning of the integration is that the response signal is less stable at the beginning of the response, therefore, by waiting a proper amount of time we can avoid such instability. It is also worth mentioning that only 32% of the response signal is used to calculate the inductance and resistance values which leave the rest of the output signal for other signal processing computations, such as offset compensation. Increasing this 32% region did not show any improvement in measurements.

Once the integrator blocks receive the enable signals, they start integrating as long as the enable signals are high. Once the inductance and the resistance are calculated the processing module forwards them to another part of the system for further analysis such as debugging and calibration. Finally, the distance module receives the inductance value and computes the distance. Distance measurement method is described in detail in the next subsection. The following pseudocode shows the inductance and resistance measurements.

Listing 1: Pseudocode for inductance and resistance measurements

```
Start
set Distance to 0.00 mm
Initialize N,Σ1,Σ2 = 0
Initialize L_avg_sum = 0
Initialize R_avg_sum = 0
repeat:
    L_avg_sum = 0
    R_avg_sum = 0
    N = 0
    repeat:
        Σ1,Σ2 = 0
        wait until t == 0.69 ms
        repeat:
            Σ1 = Σ1 + ADC_value
        until t == 1.09 ms
        wait until t == 1.69ms
        repeat:
            Σ2 = Σ2+ ADC_value
        until t == 2.09 ms
        ΣD = Compute(Σ1,Σ2,α)
        L = Derive_Inductance(ΣD)
        ΣT = Compute(Σ1,Σ2,β)
        R = Derive_Resistance(ΣT)
        store L,D to a csv file
        N = N + 1
        L_avg_sum = L_avg_sum + L
        R_avg_sum = R_avg_sum + R
        until N == 128
    L_avg = L_avg_sum/N
    R_avg = R_avg_sum/N
    store L_avg, R_avg to a csv file
    increment Distance by 0.1 mm
until Distance == 5.0 mm
End
```

We start by positioning the motorized stage to 0.00 mm in order to initialize the distance between the sensor head and the target. Then we initialize our local variables, N tracks the number of samples which is 128 for averaging the values per each distance. The averaging is done to reduce the noise in the measurements. To explain further, as the measured inductance values will be stored in a look-up table for calibration purposes, it is practical to reduce the noise as much as possible. By trial and error, we found N = 128 is a reasonable value. The sum of inductance values and resistance values are stored in L_avg_sum and R_avg_sum respectively. We then wait for 690 μs to reach the start point of $\Sigma 1$ integration. Then we integrate the ADC values until we reach the end of the $\Sigma 1$ region of interest. We repeat the same procedure to accumulate $\Sigma 2$. Once $\Sigma 1$ and $\Sigma 2$ are integrated we can calculate Σ_D by our computer function which takes $\Sigma 1$, $\Sigma 2$ and α as arguments and then by using Σ_D we can derive the inductance value L. Same principle applies to calculate Σ_T in order to derive the resistance value, R.

Then we store these raw values of L and R components to a computer using .csv file format to build up the reference look-up table for distance computation and also for further analysis such as mean/max error. We keep repeating the same computations until we reach 128 samples. Afterward, we compute the average values of inductance and resistance and store it in a .csv format. Finally, we increment the distance between the sensor and the target by 0.1 mm. We repeat the same procedure until we reach our maximum distance, i.e. 5.0 mm. by the end of this experiment, we will have 51 inductance/resistance measurement.

3.3.3. Distance Evaluation Method

Given the 51 data points of (L , D) pairs (L for inductance and D for Distance), a viable computation method is needed to compute the distance in real-time from any arbitrarily measured inductance L. The distance measurement pseudocode is shown in the following listing.

Listing 2: Distance measurement Pseudocode

```
Start
wait until t == 0.69ms
repeat:
  Σ1= Σ1 + ADC_value
until t == 1.09ms
wait until t == 1.69ms
repeat:
  Σ2= Σ2+ ADC_value
until t == 2.09ms
ΣD = Compute(Σ1,Σ2,α)
L = Derive_Inductance(ΣD)
measured_distance = look_up_and_estimate(L)
Transfer measured_distance to other components in
the design
End
```

The principle is similar to listing 1 pseudocode, however, the averaging part and resistance computations are not needed anymore as we just need the inductance value to estimate the distance in real-time. A new function called look_up_and_estimate is added to estimate the distance based on the inductance L. This function is explained in the next sub-section.

a) Inductance Interpolation

Once we gathered all 51 inductance values from the calibration process, a curve-fitting function can be applied to estimate the distance based on the measured inductance values. This function can be implemented using either a polynomial interpolation or a spline interpolation. We used cubic spline interpolation to fit our (inductance, distance) data points. We chose cubic spline interpolation over the polynomial interpolation as it showed less estimation error for our sampled experimental data. However, implementing the complete curve fitting function in the FPGA is both resource consuming and complex. Therefore, we used a combination of look-up tables and linear fit estimation to estimate the distance. The interpolation is done using 256 data points interpolated within the given 51 (L , D) sampled data points. These 256 data points are then used to form the look-up table used inside the FPGA to estimate the distance. This interpolation table is shown in Table 8 in Appendix section. This table has 256 rows with the following explanations: index is used to indicate which interpolated row we are referring to for each inductance-distance pair. DeltaDist shows the distance difference of two consecutive distances in the table. This value is used by the FPGA to estimate the distance between two consecutive inductance values in the look-up table. Moreover, these values are scaled to avoid undesirable floating point computations. 32-bit data width is used to store the necessary information of this table in the FPGA's BRAMs. The 16 higher bits of the data contains the distance value and the 16 lower bits contains the DeltaDist value.

b) Distance Calculation

Once we have the 256 interpolated data points stored in a Block RAM, we can simply look up the inductance value to estimate the distance. However, 256 discrete values cannot cover the whole range of inductance values. Therefore an estimation is needed between any two consecutive data points. This estimation is done assuming that, between two consecutive inductance data points, the behavior of the curve is linear. That being said, for any given inductance value L_{arb}, the look_up_and_estimate function first finds the nearest smaller (L,D) pair, then using the difference between L and L_{arb} as a proportion, it estimates the distance.

3.4. Experimental Results

3.4.1. Inductance Measurement

After implementing the inductance measurement method in the FPGA, now we are able to perform a set of inductance measurement readings at different distances. These readings are carried out in the range of 0.00 mm and 5.00 mm with the granularity of 0.1 mm and are averaged over 128 samples at each distance for a total of 51 distances. The set distance is controlled by a TCL script that relays the programming commands to the motorized stage to reach the highest accuracy and measurement repeatability. A set of averaged inductance readings for a Crouzet sensor is shown in Table 2 and plotted in Figure 20.

Table 2: Averaged Inductance values for a Crouzet sensor

Set Distance [mm]	Inductance [mH]	Set Distance [mm]	Inductance [mH]
0	52.15083	**2.6**	24.11484
0.1	46.70989	**2.7**	23.98133
0.2	42.34707	**2.8**	23.8583
0.3	39.25833	**2.9**	23.74597
0.4	36.92822	**3**	23.64037
0.5	35.05347	**3.1**	23.5425
0.6	33.53058	**3.2**	23.45369
0.7	32.26843	**3.3**	23.37042
0.8	31.1981	**3.4**	23.29344
0.9	30.29268	**3.5**	23.22267
1	29.49453	**3.6**	23.15644
1.1	28.80336	**3.7**	23.09489
1.2	28.20017	**3.8**	23.03702
1.3	27.66535	**3.9**	22.98496
1.4	27.19682	**4**	22.93339
1.5	26.77032	**4.1**	22.88852
1.6	26.39188	**4.2**	22.84352
1.7	26.05323	**4.3**	22.80414
1.8	25.7483	**4.4**	22.76758
1.9	25.47293	**4.5**	22.7309
2	25.21949	**4.6**	22.69883
2.1	24.99116	**4.7**	22.66901
2.2	24.78253	**4.8**	22.63863
2.3	24.59249	**4.9**	22.61136
2.4	24.42145	**5**	22.58637
2.5	24.26161		

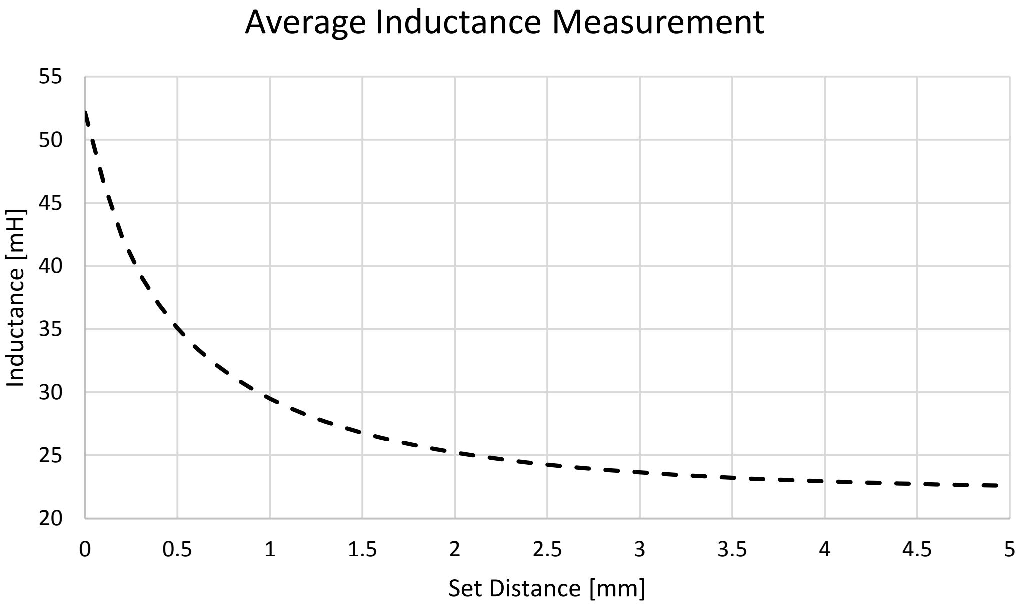


Figure 20: Averaged Inductance Measurements plot for a Crouzet sensor

In Figure 20, we observe a decreasing trend in inductance when the distance increases. We can also see that the variation of inductance is higher in the range of 0.00 mm and 3.00 mm and the slope decreases significantly after 3.00 mm. This is due to the characteristic of the sensor as it is more sensitive when the target moves within the closer range of it. In the next section, we investigate this behavior and see how it affects the distance measurement. The maximum inductance value is 52.15 mH, which occurs at 0.00 mm and the minimum inductance value is 22.59 mH, which occurs at 5.00 mm.

The inductance measurements are conducting in real time using the FPGA and they are prone to errors due to the noise. Therefore, it is important to classify these errors and draw meaningful results from it. We used mean error over 128 readings at each distance to see how much inductance variation there is from the average inductance value.

Figure 21 shows the mean error for the Crouzet sensor. The corresponding errors for all 51 distances are shown in Table 7 in the Appendix section. These results have been analyzed over 128 samples at each distance. As we see in Figure 21, the mean error is less than 0.008 mH (8 µH) for the Crouzet sensor. Moreover, we observe that the inductance measurement error is independent of the distance and has almost the same value over all distances in this set of measurements.

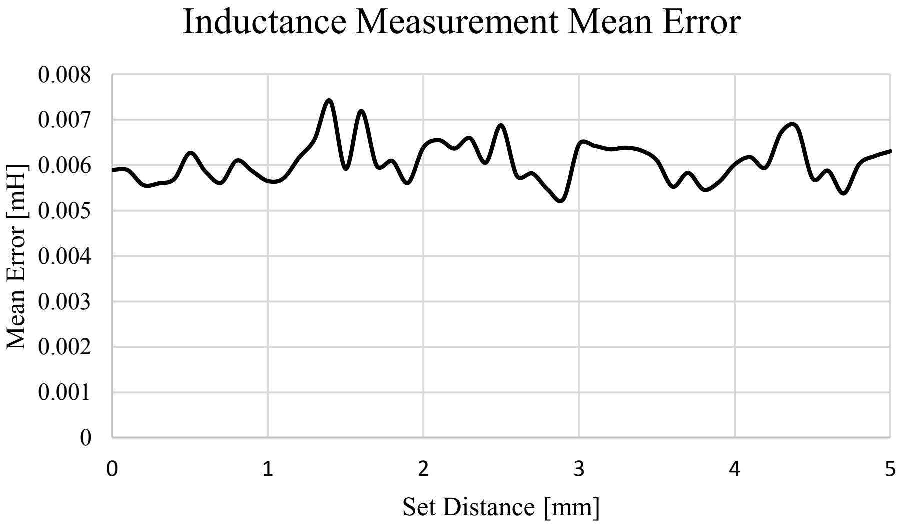


Figure 21: Inductance Measurement Mean Error for Crouzet sensor

3.4.2. Distance Measurement at Room Temperature

The distance measurement is done using the FPGA, for reaching the highest accuracy each sensor needs to be calibrated before running the distance measurement test. Calibration is the process of reading the inductance values of the sensor in order to generate the interpolation table and update the distance BRAM in the FPGA. The movement of the sensor along with capturing data and storing them on a computer is fully automated using TCL and Python scripts. After performing the inductance data acquisitions, raw data are used to build and calibrate the necessary look-up tables and coefficients factors using a Python script. Each calibrating operation performs 51 set of readings for different distances from 0 mm to 5 mm with the granularity of 0.1 mm. The automation process of target movement and data acquisitions are explained in detail in chapter 5. In this section, we investigate the distance measurement for the Crouzet sensor and analyze its measurement error at different distances.

Figure 22 shows the mean error for measured distance for the Crouzet sensor by the FPGA after calibration. They have been measured over 128 samples for each distance, from 0.00 mm to 5.00 mm with the granularity of 0.1 mm.

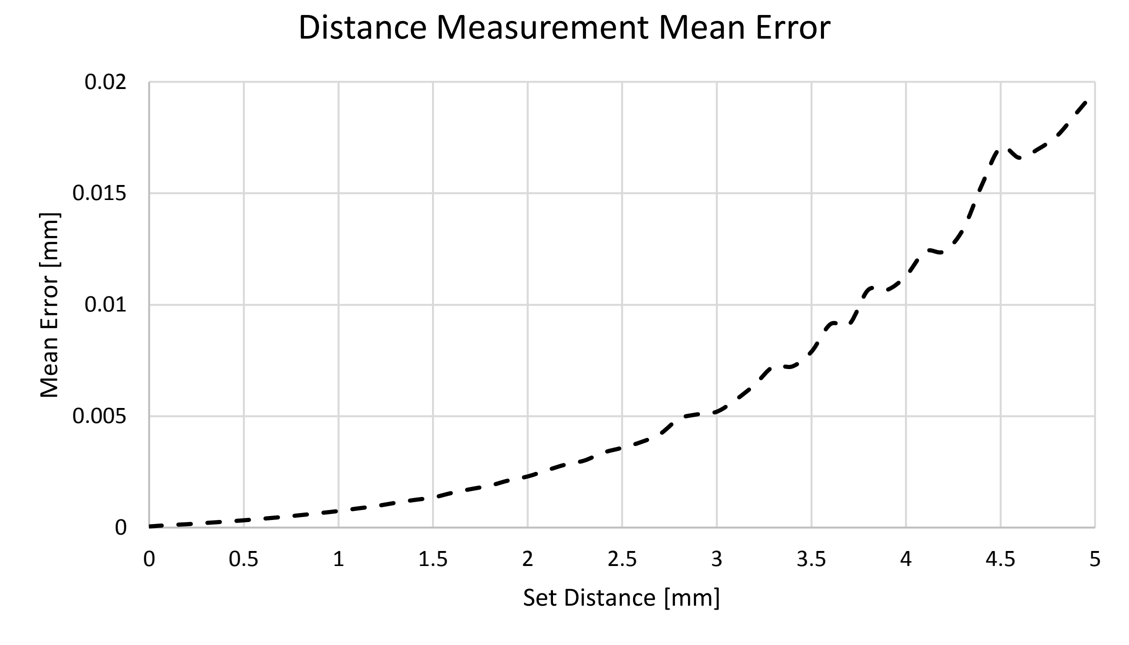


Figure 22: Distance Measurement Mean Error for Crouzet Sensor

It can be observed that error increases more rapidly above 3.00 mm. This is expected as the inductance itself does not change significantly above 3.00 mm, therefore makes it harder to distinguish the distance in the range of 3.00 mm and 5.00 mm. The maximum mean error for the Crouzet sensor occurs at 5.00 mm which is 0.02 mm or 1%. This defines our confidence level of our measurement accuracy using a Crouzet sensor. In the next chapter, we will compare the Crouzet sensor with other sensors as well to see whether they have an error rate less than 2% or not. For a better illustration, Figure 23 shows the raw error for the Crouzet sensor for all 128 samples at each distance. The maximum error occurs at 5.00 mm which is 0.075 mm which is 1.5 %.

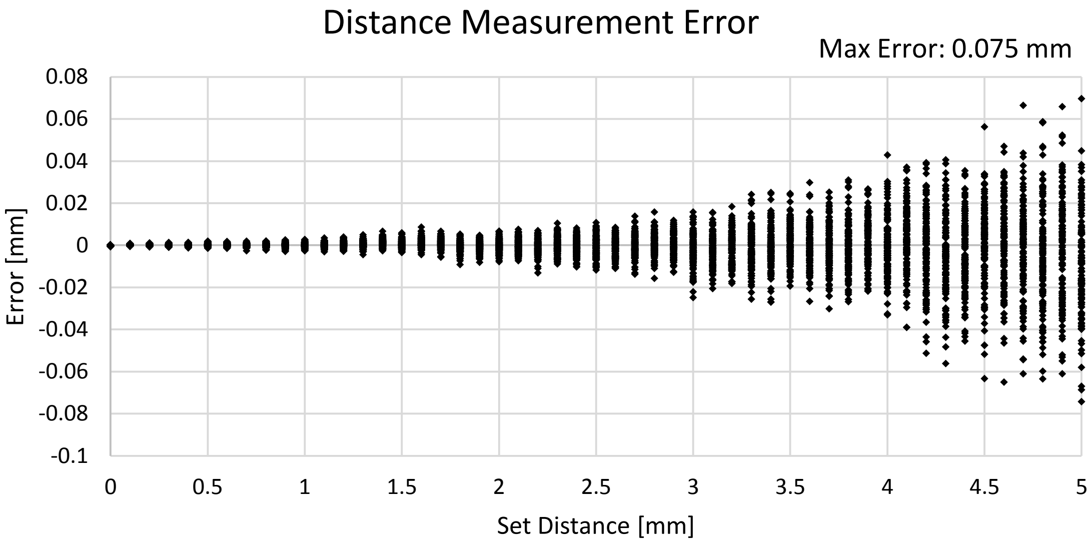


Figure 23: Error Measurements for Crouzet

3.4.3. Distance Measurement at Different Temperatures

One of the main motivation of using the current ramp method was compensating the temperature drift automatically with minimum computational overheads by separating the inductance and resistance in the first place. In order to see the error, we examined our proximity measurement system at three different temperatures: -30 °C, 25 °C (room temperature) and +70 °C. The distance mean error of a Crouzet sensor at these temperatures are shown in Figure 24.

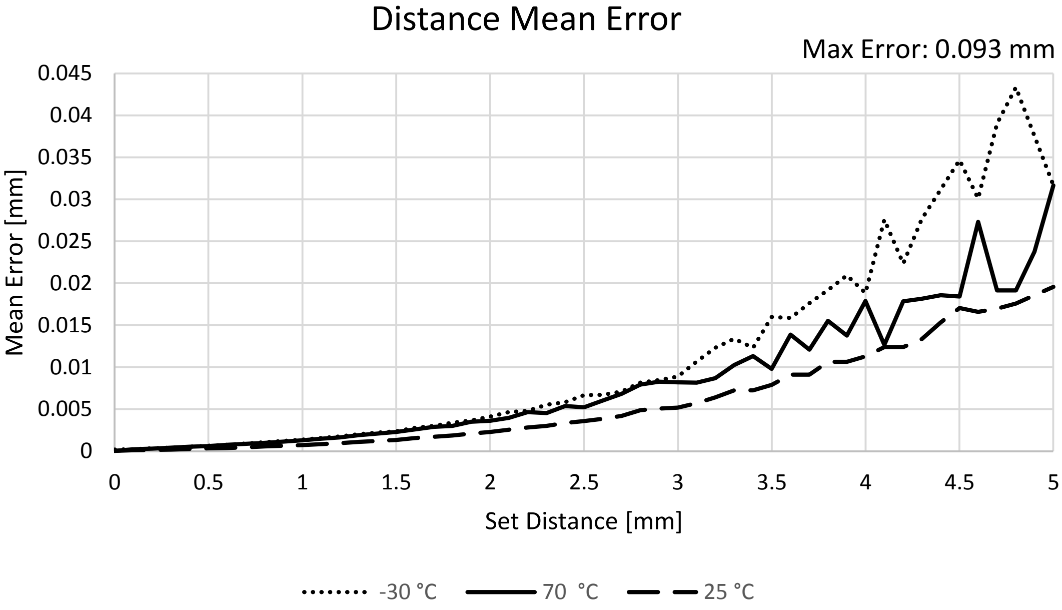


Figure 24: Mean Error for Crouzet Sensor at different temperatures

We observe that both -30 °C and +70 °C have higher error rates in comparison to room temperature. However, it is still within our requirements which was less than 2% distance measurement error at all distances. The maximum error occurs at 5.00 mm which is 0.093 mm and it is less than 2%.

3.5. Summary

In this chapter, we described the theory behind the current ramp excitation method and its computational advantage over the previous methods. Moreover, we explored the hardware implementation of the current ramp method using the combination of look-up tables and linear fit algorithm. We also performed a wide variety of experiments at different temperatures and concluded that the Crouzet sensor has less than 1.5% error at room temperature and less than 2% error at extreme temperatures. In the next chapter, we explore inductive proximity sensors with different characteristics. Moreover, we compare their accuracy performance and see which one works better with our current ramp method.

Chapter 4

Design Extension to Support Multiple Proximity Sensors

While all the inductive proximity sensors are built using a coil as their primary transducer, not necessarily all the sensors have the same internal characteristics. In other words, the operational inductance range can differ from one manufacturer to another. Therefore, each manufacturer has to design and build its own electronic circuits to support its sensors. As a result, the integration of different electronic circuits in one system can be quite costly. In this chapter, we extend the previous design explained in chapter 3 to support multiple sensors using a single processing circuit. This approach can significantly reduce the cost. In order to support multiple sensors, we need to add scalable features to our design, such as flexibility of choosing among sensor types as well as being able to efficiently multiplex among them. The sensors that we chose for our system are four commercial off-the-shelf inductive proximity sensors used for avionic applications: Crouzet, Goodrich, Crane and Honeywell. FT84798 from Crouzet [12], 40180 from Goodrich [13], 8-642 from Crane [20] and 320FW04-5 from Honeywell [14]. We start by analyzing the different characteristics of the chosen sensors, then we explain how to scale our design to support up to 10 sensors. Moreover, we run different tests to compare the accuracy of these sensors, both at room temperature and extreme temperature ranges.

4.1. Characteristic Analysis of Different Sensors

As discussed earlier, each sensor can have its own inductance and resistance ranges. The minimum and maximum inductance values, L_min and L_max have been shown in Table 3 for four sensor types. This table shows that the inductance range varies from sensor to sensor. Moreover, it shows that they all have the same decreasing trend for the inductance value when the distance increases.

Table 3: Inductance range of different sensor types

	Inductance (mH)			
Sensor Type	Crouzet	Honeywell	Crane	Goodrich
0 mm (L_max)	52.15	20.19	5.92	5.85
5 mm (L_min)	22.6	13.07	4.94	4.52

Furthermore, as we have seen in section 3.4.1 the inductance mean error is independent of the distance. Therefore, we used the average mean error over all distances to compare the inductance mean error among different sensors. The more precise we measure the inductance, the more accurate we can measure the distance. However, it should be noted that just considering the mean error would not necessarily lead to a fair comparison, as the inductance range differs among sensors, therefore a more fair comparison is given when dividing the mean error to the inductance range for each sensor. These results are shown in Table 4.

Table 4: Inductance Mean Error Comparison among sensors

Sensor Type	Goodrich	Crane	Honeywell	Crouzet
Mean Error (mH)	0.001595	0.000594	0.002868	0.006083
Mean Error /L_{range} (%)	0.119924	0.060612	0.040312	0.017079

The Crouzet sensor shows the minimum mean error divided by the inductance range among other sensors and as we will see in the following sections, this property increases the distance measurement accuracy. In the next section, we will see how to create look-up tables for different sensors.

4.2. Look-up Table Creation

As we have seen in the previous section, the inductance range for each sensor type is different. Therefore, each sensor type needs its own interpolated inductance look-up table to estimate the distance correctly. Having four sensor types leads to four rom pages in order to separately store the look-up tables. 32-bit data width is used to store these values which means for each sensor type we need 256×32-bit values which make a total of $4 \times 256 \times 32$ bits to cover all the sensor types. This makes a total of 32 Kbits which can be simply stored in a 36 Kbits block RAM.

4.3. Sensor Multiplexing Design

Once the look-up tables are created, a proper multiplexing policy needs to be applied to choose among 10 sensors. Figure 25 shows the abstract diagram of the multiplexing system, for simplification, only three sensors out of ten have been shown.

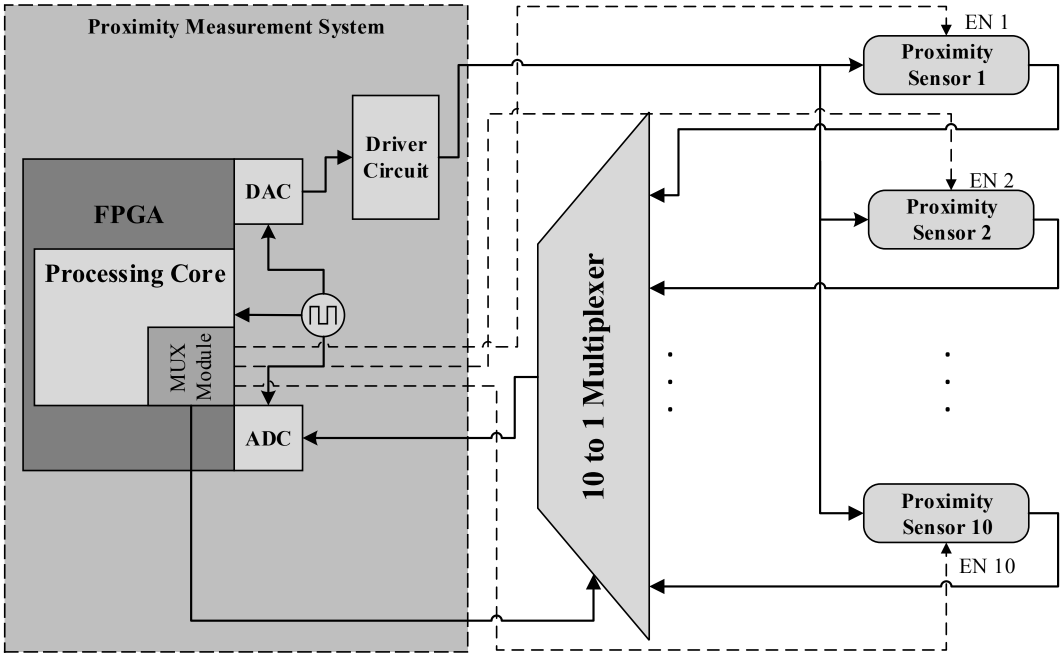


Figure 25: Proxy Measurement System Supporting up to 10 Sensors

The output of the driver circuit is now extended to all the proximity sensors. However, the current does not flow to any proximity sensors until its corresponding enable bit is asserted. The MUX module inside the processing module is responsible for choosing the appropriate enable bit to be asserted as well as selecting the right sensor's output through the multiplexer to be forwarded to the ADC. For example, if it is the 2nd proximity sensor's turn, then only the enable bit number 2, i.e. EN 2 signal is asserted and all the other enable bits are disabled. On the multiplexer side, the mux selector which is controlled by the MUX module, chooses the output of the 2nd proximity sensor to be forwarded to the ADC. Considering a period of 2.5 ms for excitation, 25 ms needed to poll all the sensors. That also means each sensor updates its distance with a refresh rate of 40 Hz. Moreover, only one sensor at a time is excited to reduce the power consumption.

4.4. Experimental Results

In this section, we start by comparing the measurement accuracy among four sensor types at room temperature using our extended current ramp method. We also report what the maximum error for each sensor type is. Then we move forward comparing them at extreme temperature ranges.

4.4.1. Measurement Error Comparison at Room Temperature

Figure 26 compares the mean distance error among the four chosen sensor types at room temperature.

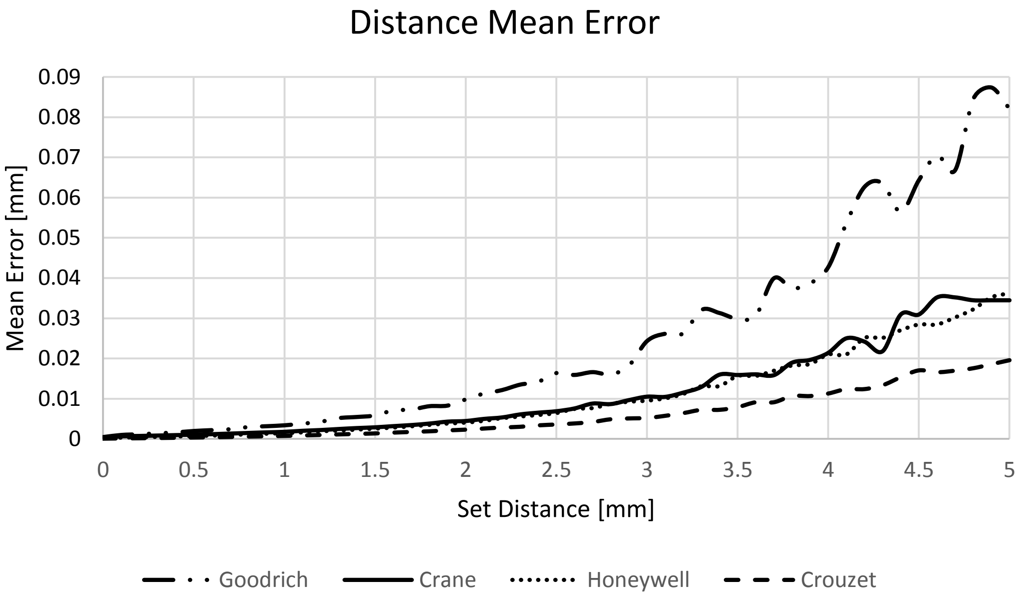


Figure 26: Mean Error Comparison at Room Temperature

The mean error increases for all sensor types as distance increases. This is expected as all sensors have more sensibility for the range of 0 – 2 mm. The reason that the Crouzet sensor has the minimum mean error among other sensors is that the inductance mean error over the total inductance range is lower in comparison to other sensors. Therefore, each distance has more space to be separated to its consecutive distances.

As we observe, the Crouzet sensor shows the minimum mean error which is less than 0.02 mm and the Goodrich Sensor shows the maximum mean error which is about 0.09 mm. The Honeywell sensor and the Crane sensor behave almost the same with the mean error of less than 0.04 mm. This is expected, as the Crouzet sensor has the largest inductance range among other sensors. The maximum error for all the sensors at room temperature has been illustrated in Figure 27.

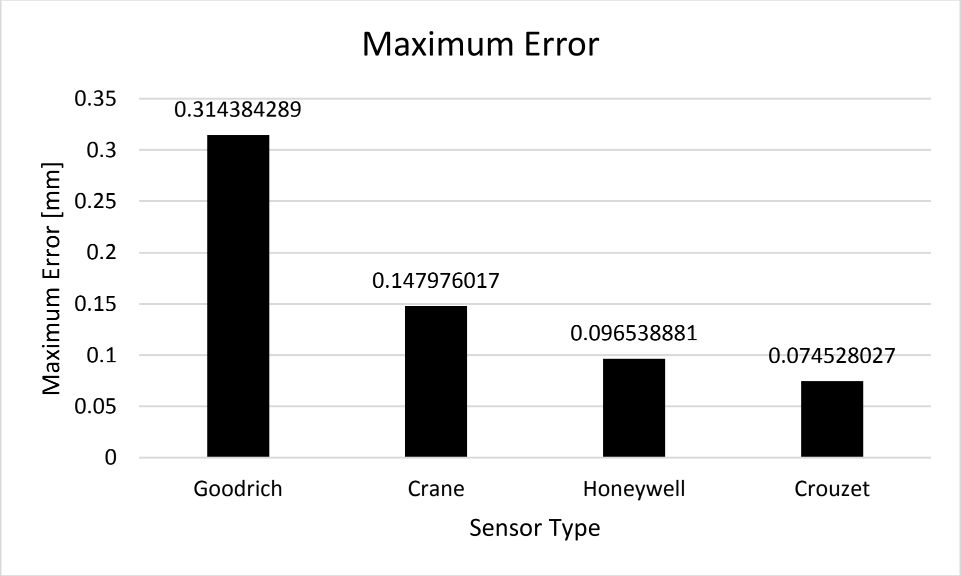


Figure 27: Max Error Comparison at Room Temperature

4.4.2. Measurement Error Comparison at Extreme Temperature Ranges

The biggest advantage of current ramp method is its ability to automatically compensate the temperature drift with the minimum amount of computational overhead by separating the inductance and resistance components in the first place. In order to observe the error, we examined our proximity measurement system at three different temperatures: -30 °C, 25 °C (room temperature) and +70 °C. The distance mean error of all sensors at these temperatures are shown in Figure 28 to Figure 31.

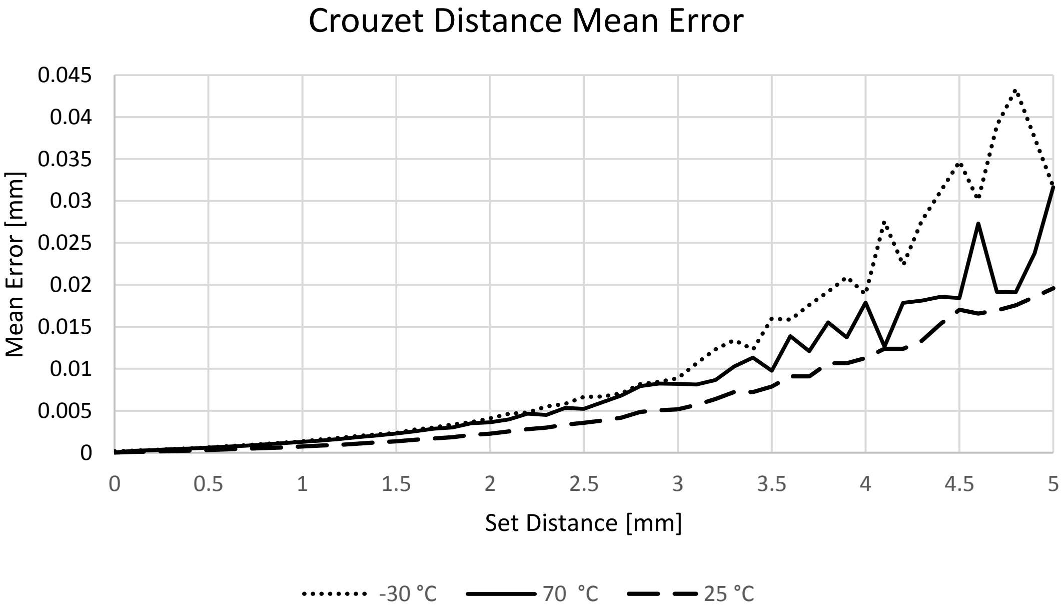


Figure 28: Distance Mean Error for Crouzet sensor at different temperatures

]

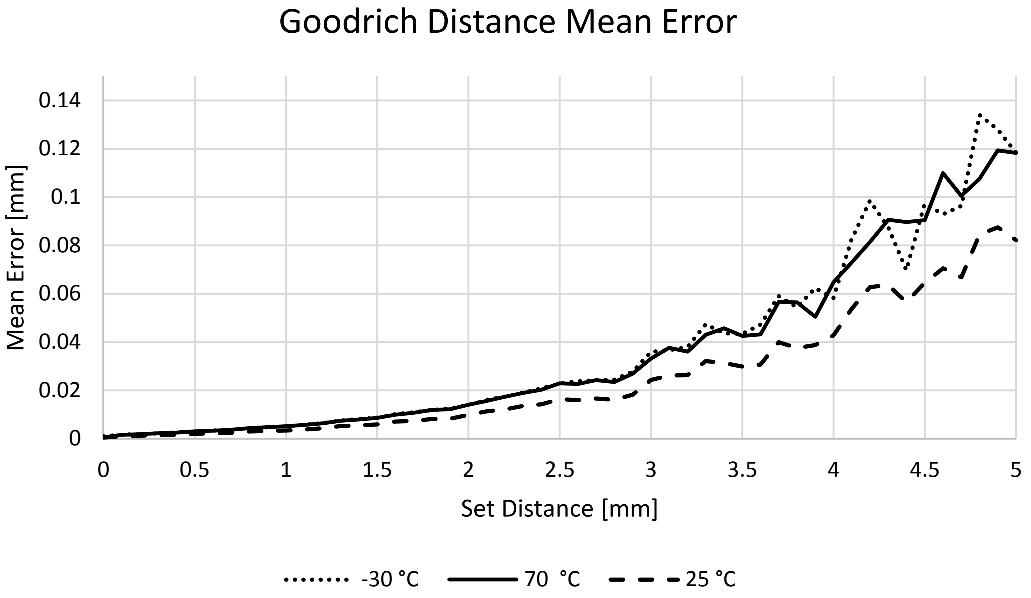


Figure 29: Distance Mean Error for Goodrich sensor at different temperatures

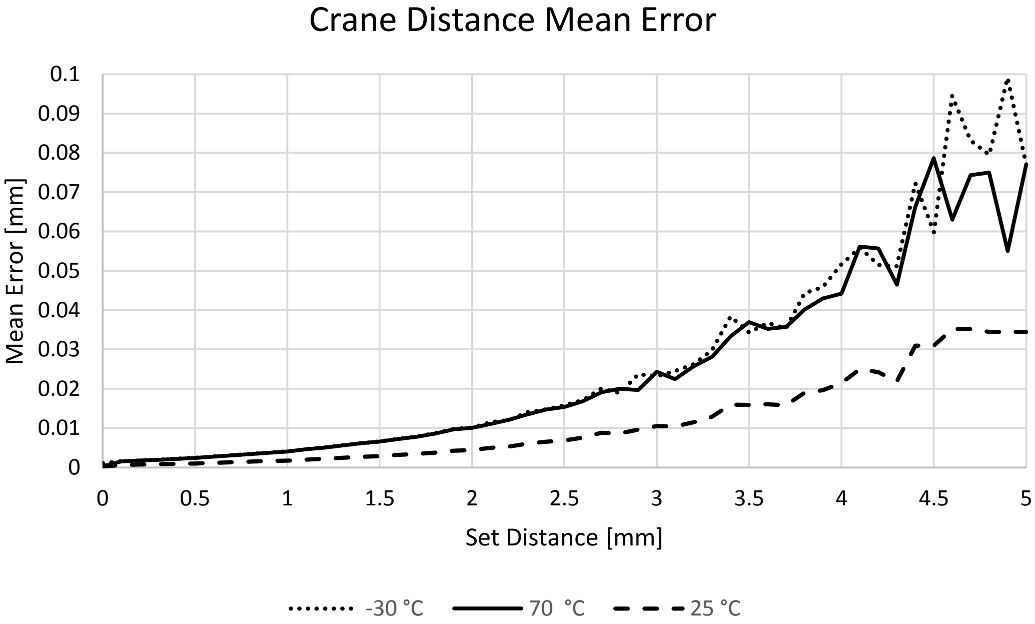


Figure 30: Distance Mean Error for Crane sensor at different temperatures

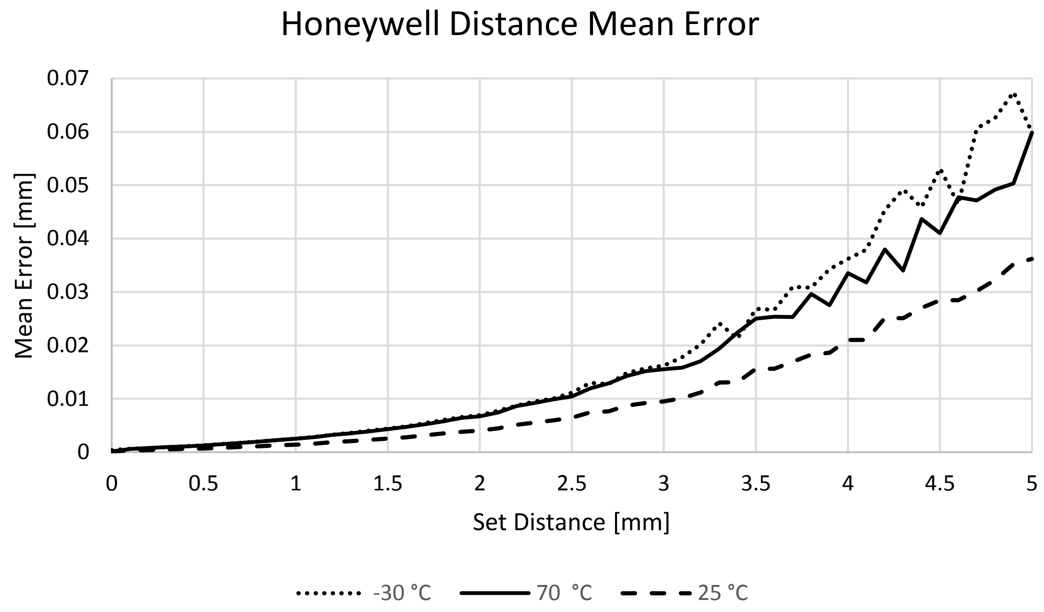


Figure 31: Distance Mean Error for Honeywell sensor at different temperatures

The results prove that current ramp excitation method can reduce the temperature drift in a wide variety of inductive proximity sensors used in aerospace industry. However, worst case scenarios need to be examined as well. The maximum error at extreme temperature ranges for all four sensor types is shown in Table 5.

Table 5: Maximum Distance Error Comparison Among Sensors

	Maximum Error [mm]			
Temperature [°C]	Goodrich	Crane	Honeywell	Crouzet
25	0.314384289	0.147976017	0.096538881	0.074528027
-30	0.354865483	0.212615083	0.128784242	0.093497787
70	0.350571244	0.190544996	0.120201683	0.090635606

4.5. Summary

In this chapter, we explained how to extend our design to support multiple sensor types with different internal characteristics. Furthermore, we discovered each proximity sensor has its own specific inductance range, thus proper look-up tables need to be created for each sensor type. Moreover, we compared four different sensor types at room temperature and extreme temperature ranges. The Crouzet sensor shows the most accurate result with the worst case error of 0.75 mm which is less than 2% distance error. The Goodrich sensor shows the worst results with the worst case error of 0.35 mm which is about 7%. In the next chapter, we explore the test and development automation. We also illustrate the complete proximity measurement platform that includes the test platform as well. We then talk about the detailed implementation of our automation scripts.

Chapter 5

Test and Development Automation

Once the proximity measurement system is designed, it is crucial to test it efficiently and completely. Especially in the avionic industry that the safety and the security of the passengers come first. These tests are often repetitive and time-consuming. Therefore, designing a proper test platform not only reduces the development cost, but also it prevents any unforeseen experimental errors that may be carried out by a human.

In this chapter, we start by designing an automated test platform that integrates the software and hardware components, then we explain our test methodology. Moreover, we investigate our hardware/software components, then we move forward explaining Python and TCL scripts that developed to perform the test procedures as well as accelerating the FPGA design by automating the repetitive manual procedures in the design.

5.1. System Overview

Figure 32 shows the complete proximity measurement system along with the test platform. It should be noted that the small components are also included in this figure which will be explained briefly in this section. Broadly speaking, three main components are available on our platform. These three components are:

1. A **Proximity Measurement System** that interfaces with the inductive proximity sensors and performs the inductance, resistance and distance measurements.

2. The **Inductive Proximity Sensor** which resides on top of the linear motorized stage, providing the flexibility of choosing the sensor to target distance.

3. A **Computer** that communicates with the FPGA and the motorized stage as well as running the calibration and test scripts.

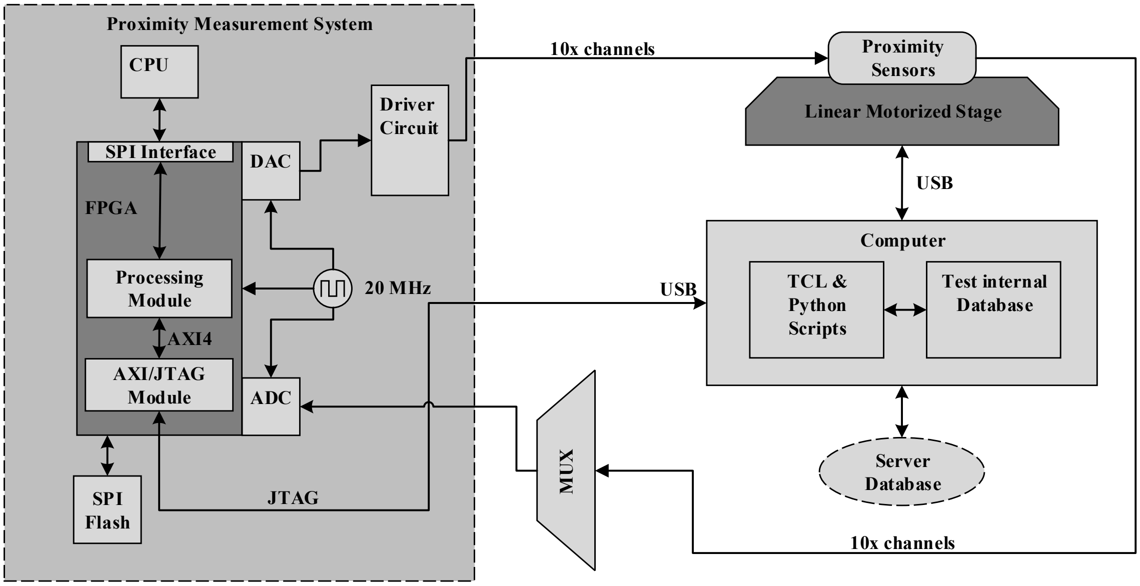


Figure 32: Proximity Measurement System along with the Test Platform

The proximity measurement system receives information from proximity sensors at a different location in the aircraft (e.g. doors, landing gear). Then, after processing the raw information, it extracts the desired system requirements such as distance to send them to a higher level of the system (i.e. CPU). Proximity measurement System accommodates

FPGA, SPI Flash Memory, Driver Circuit, ADC/DAC and CPU. FPGA runs the hardware implementation of the measurement system and it is also responsible for controlling various components such as DAC (Digital to Analog Converter) to generate the excitation pattern and ADC (Analog to Digital Converter) to receive the response signal in a digital format.

The linear stage is acquired to accurately position the distance between the sensor and the target. It is also used to automate the calibration and test procedures. This linear stage is controlled by TCL/Python script resides on the computer. Moreover, the computer is responsible for the direct programming of the FPGA as well as configuring the SPI Flash. Furthermore, it is used to run the test and automation scripts for the real-time communication between the computer and the FPGA. SPI Flash Memory uses the Serial Peripheral Interface (SPI) to configure the FPGA. It contains the hardware program for the FPGA and each time the power of the system cycles, it loads the configuration file to program the FPGA. The JTAG/AXI block provides a bidirectional communication between the FPGA and the computer.

5.2. Motorized Linear Stage

As mentioned earlier, the motorized linear stage is used to accurately position the distance between the sensor and the target. The motorized stage that we used is MFA-CC from Newport [21]. MFA-CC provides precision motion with a travel range of 25 mm powered by a DC motor. The guaranteed on-axis accuracy is ±3 μm with a guaranteed uni-directional repeatability of 0.3 μm. These features along with the ability to program the movement of the stage, make it suitable for our test platform. Figure 33 shows the MFA-CC stage.

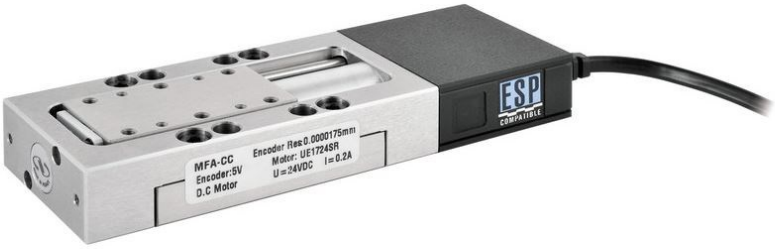


Figure 33: Motorized linear stage

The inductive proximity sensor along with the target needs to be mounted on this stage. A customized jig is built and assembled for this purpose. Figure 34 shows the stage alongside with the sensor and the target mounted on top of it. The motorized linear stage can receive commands from a computer through a serial port interface. These commands are used to control the motion of the stage and can automate the target's movement using any scripts capable of establishing a serial communication. We have used a TCL script for such communication.

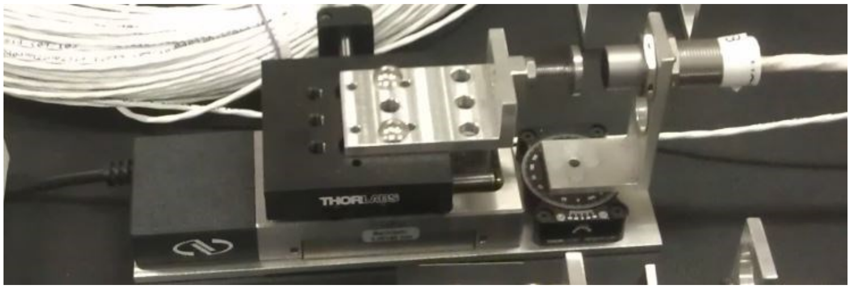


Figure 34: Motorized linear stage with sensor and target

5.3. JTAG/AXI Block

Once we measured the inductance values and resistance values with the FPGA, it is important being able to transfer them to a computer for further analysis and debug purposes. JTAG/AXI block is a bridge to establish a communication between the computer and the FPGA. This is a customizable core that can perform AXI transactions such as read and write transactions. JTAG to AXI Master can drive the internal AXI signals in the FPGA using the write transaction as well as reading the values using the read transactions. The communication protocol can also be customized using either AXI4 protocol or AXI4-lite protocol. The AXI data-width is also flexible and it can be chosen either as a 32-bit data-width or 64-bit data-width.

The AXI/JTAG block is fully designed and tested using Xilinx IP Integrator tool from Vivado Design suite. IP Integrator allows you to instantiate your computing building blocks using a simple drag and drop method and making communication between them. After making all the instantiation and connection this block can be instantiated in the design as a standalone IP. A block memory is used to store the inductance and resistance values. Once the write transaction is performed, the block memory is erased and then initiates the data acquisitions from address 0. After capturing 128 values of inductance and resistance, a series of burst read transaction are performed to transfer the data from the block memory to the computer in a .csv file format. Write and read transactions are described in the next sub-sections.

5.3.1. Write Transaction

Write transaction initiates the data acquisitions as well as selecting which proximity sensor channel to capture the data from. We used a 64-bit data-width write transaction to transfer the appropriate data from the computer to the FPGA. First, the write transaction needs to be created using the following TCL command:

```
create_hw_axi_txn gpio_write64 [get_hw_axis hw_axi_1] -type write –data\
{00000000_00000000} -address  $gpio_addr -size 64
```

-type specifies whether it is a read or write transaction. –data specifies the initial value of the transaction. -address defines in which address space the transaction needs to be performed. This is because there might be other JTAG to AXI Master Modules available in the design with different address spaces. Finally, the -size option chooses the data-width which in our case is 64. Once the transaction is created, the write_data property can be set according to which sensor channel needs to be selected. This can be done using the following TCL command:

```
set_property DATA $write_data [get_hw_axi_txns gpio_write64]
```

It should be noted that there is no predefined data format available for the write_data property. The data format is conventional and it has to be defined by the designer and interpreted correctly by creating necessary computing blocks in the. AXI/JTAG block. Finally, we need to run the actual AXI transaction using the following command:

```
run_hw_axi [get_hw_axi_txns gpio_write64] –quiet
```

5.3.2. Read Transaction

When 128 data samples are captured in the block memory, we have to read and store them on the computer, however as these data set can be quite big, it is faster to do a burst read. The burst lengths can be set up to 256 which mean 256 reads in one transaction. Considering 64-bit data-width, it is a $256\ \times 64 = 16\ Kb$ per transaction. Read transaction can be created using the following TCL command:

```
create_hw_axi_txn read256x64 [get_hw_axis hw_axi_1] -type read -len 256 \
-address $base_address -size 64
```

-len defines the burst size and the other options are already explained in the write transaction.

5.4. Script Development

TCL scripts along with Python scripts have been developed to accelerate the testing procedures as well as automating some part of the design. Python scripts are mainly involved for automating the offline numerical computations of the design such as performing spline interpolation. TCL scripts, on the other hand, perform real-time communication among different platforms to integrate everything together.

5.4.1. Calibration Script

When we install a new sensor to the proximity measurement system, it needs to be calibrated. The calibration process is vital for reaching the maximum accuracy and reliability level. The general idea of calibration is putting the sensor at different distance positions, and measuring its inductance, then creating an interpolated table based on these inductance readings. However, this procedure can be time-consuming if it is done manually. For example, a manual scenario would be setting the distance at the beginning, then reading 51 inductance values and putting them in an excel sheet. Then by interpolation values, creating a distance table, transferring it to the BRAM and then recompiling the FPGA to see the effect. By automating this process a huge amount of time can be saved, therefore the development and test cost can be reduced significantly. We used TCL script to serve this purpose. Listing 3 shows the calibration script pseudocode.

Listing 3: Calibration Script Pseudocode

```
Start
Initialize HW_AXI_read
Initialize HW_AXI_write
Initialize Motorized_Linear_Stage(serial_port)
Create sensor_avg.csv
Create axi_write_store_tnx(sensor_channel)
Create axi_write_ram_rst_txn
Create axi_read_data_txn
Get_Input(sensor_specs[])
Get_Input(sensor_channel)
repeat:
   HW_AXI_write(axi_write_ram_rst_txn)
   Motorized_Linear_Stage.move(set_distance)
   HW_AXI_write(axi_write_store_tnx)
   wait(8s)
   axi_read_data = HW_AXI_read(axi_read_data_txn,HEX)
   sensor_readings[] = axi_read_data.extract()
   R_avg = sensor_readings[r_avg]
   L_avg = sensor_readings[l_avg]
   store.append (L_avg,R_avg,-sensor_avg.csv)
   increment set_distance by 0.1 mm
until set_distance == 5.0 mm
Py_exec DB_update.py(sensor_specs[],distance_table.coe)
distance_table.coe = Py_exec make_intrp.py(sensor_avg.csv)
Py_exec BRAM_update.py(Distance_BRAM_update,distance_table.coe)
FPGA.compile()
FPGA.program()
End
```

First, we initialize the AXI read and write hardware as well as initializing the motorized stage by specifying the specs for the serial interface. Then a .csv file is created to save the average values of inductance and resistance. Furthermore, two write AXI transactions along with a read AXI transaction are created to perform the write and read transactions. Afterward, we receive the sensor specs from the user. This spec includes sensor type, sensor serial number, target type, etc are used to store in a database. This database can be used later to reload a characterization Bitstream into the FPGA.

Now, as we have 10 sensor channels, we ask the user to choose the appropriate channel for the active sensor. Then, we reset the content of the block RAM, this BRAM temporarily stores the inductance and resistance values to be later read by the computer. After resetting the block RAM we send a serial command to the motorized stage to set the distance. Once the distance is set, we perform an axi_write transaction, this transaction initiates the data acquisitions of inductance and resistance to the block RAM. Then we have to wait 8 seconds to store a set of 128 readings. Once the data acquisition is finished, we perform an axi_read transaction to transfer the data from the block RAM to the computer and save it to the computer rafter extracting the proper information out of that, such as averaging the inductances values over 128 samples. We repeat these steps until we reach the 5.00 mm distance which is our maximum experimental distance.

Once the data storage and extraction at different distances is finished and stored in the computer, the make_intrp.py Python script is called to perform the interpolation and generate the distance look-up table in a .coe format. The generated .coe file is then used by the BRAM_update.py script to update the FPGA’s distance BRAM. The last python script is called to update the calibration database. It is important to keep track of the sensor calibration to easily reload the calibration data in the FPGA if necessary. A database is created for that. Finally, we recompile and reprogram the FPGA to see the calibration effect. In terms of timing, the manual sensor characterization can take up to 40 minutes (excluding the FPGA compile time) while the automated sensor characterization takes less than 7 minutes.

5.4.2. Test Script

Once a sensor is calibrated it needs to be tested for performance and accuracy evolution. A TCL script has been developed to ease the test performance. The pseudocode of the test script is shown in Listing 4.

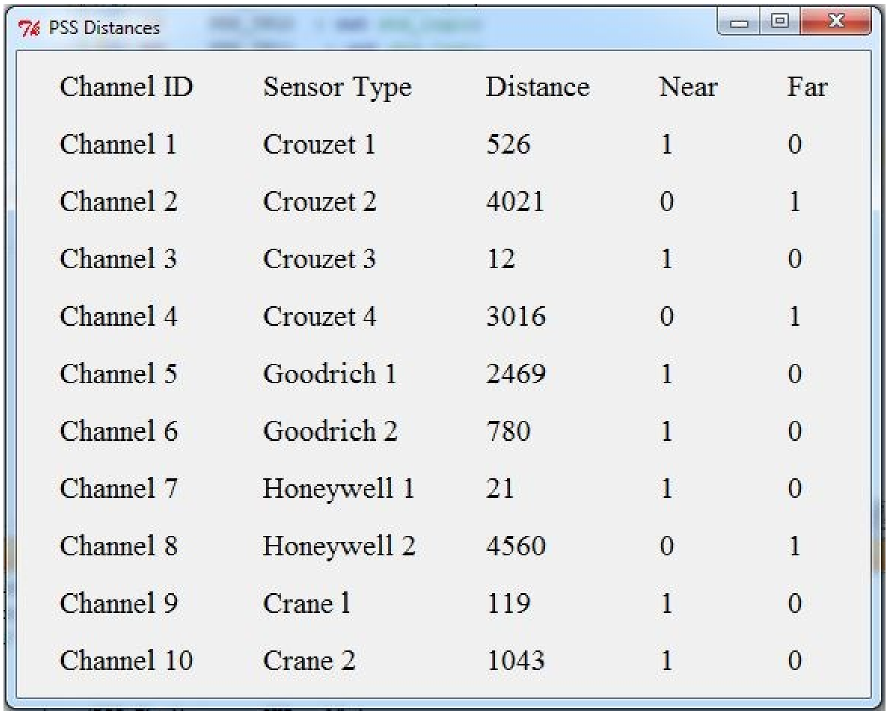
Listing 4: Test Script Pseudocode

```
Start
Initialize HW_AXI_read
Initialize HW_AXI_write
Initialize Motorized_Linear_Stage(serial_port)
Create set_distance_vector[v]
Create axi_write_store_tnx(sensor_channel)
Create sensor_avg.csv,sensor_raw.csv,sensor_analysis.xlsx
Create axi_write_ram_rst_txn
Create axi_read_data_txn
Get_Input(sensor_channel)
repeat:
   HW_AXI_write(axi_write_ram_rst_txn)
   Motorized_Linear_Stage.move(set_distance)
   HW_AXI_write(axi_write_store_tnx)
   wait(8s)
   axi_read_data = HW_AXI_read(axi_read_data_txn,HEX)
   sensor_readings[] = axi_read_data.extract()
   R_avg = sensor_readings[r_avg]
   L_avg = sensor_readings[l_avg]
   measured_distance = sensor_readings[d]
   store.append (L_avg,R_avg,-sensor_avg.csv)
   store.append (L,R,measured_distance,-sensor_raw.csv)
   set_distance = distance_test_vector.deque()
until distance_test_vector.empty()
sensor_analysis.xlsx = Py_exec make_analysis.py(sensor_raw.csv)
End
```

The principle of data acquisitions and target movement remains the same, however, a few changes need to be done in the TCL script as well as creating the proper python scripts for further analysis. A test vector is created at the beginning and is used to specify which distances are needed to be set. Further, two .csv files are created in addition to the sensor_average.csv. The first file is sensor_raw.csv which stores the raw values from the sensor readings. The second file is an analysis excel file that saves results from the python analysis script. In line 20, we also extract the measured_distance values from the readings. The measured_distance is computed by the FPGA. In line 25 we evaluate the test results by calling the make_analysis.py Python script. These analysis includes reporting standard deviation, generating error graph, etc.

5.5. GUI Development

While saving data into excel file formats such as .csv or .xlsx is necessary for performance and error analysis, it is not always convenient for real-time observation and quick tests, for example, checking a sensor's inductance/distance value. Therefore, Creating a Graphical User Interface (GUI) would be helpful to serve such requirements. Figure 35 shows a viewer which shows the distance along with its near/far status in real-time. The distances are in micrometers. The near/far status is determined by Boolean values, 1 stands for the true state and 0 stands for the false state. For example, in the first channel, we observe that the distance value is indicated by 526, which means it is at 0.526 mm and the Near attribute is indicated by 1 which means that the target is in the near range (less than 2 mm in this setup). This viewer has been developed using the TCL/Tk language. The data transfer occurs in real-time via AXI/JTAG block.



Channel ID	Sensor Type	Distance	Near	Far
Channel 1	Crouzet 1	526	1	0
Channel 2	Crouzet 2	4021	0	1
Channel 3	Crouzet 3	12	1	0
Channel 4	Crouzet 4	3016	0	1
Channel 5	Goodrich 1	2469	1	0
Channel 6	Goodrich 2	780	1	0
Channel 7	Honeywell 1	21	1	0
Channel 8	Honeywell 2	4560	0	1
Channel 9	Crane 1	119	1	0
Channel 10	Crane 2	1043	1	0

Figure 35: Distance viewer GUI

5.6. FPGA Resource Utilization

FPGAs accommodate different resources for reconfigurability purposes such as LUTs (Look-Up Tables), BRAMs (Block RAMs) etc. The availability of these resources depends on the FPGA family series as well as the specific device. In our implementation we have used a XC7A100T device from Xilinx's Artix-7 Family. Artix-7 devices offers the lowest power and cost that fits perfectly for our avionic application [22]. After implementing the design, we can evaluate the resource utilization of our design. Table 6 shows the available resources as well as resources utilization of our design.

Table 6: XC7100T Artix-7 FPGA Resource Utilization

Resource	Utilization	Available	Utilization %
LUT	12843	63400	20.26
LUTRAM	2272	19000	11.96
FF	17857	126800	14.08
BRAM	67	135	49.62
DSP	92	240	38.33
IO	95	170	55.88
BUFG	5	32	15.62

LUT refers to the look up tables. These look up tables are the building blocks of the logic design and they configure the functionality of the design. BRAM refers to the Block RAM, BRAMs provides main memory resources for the design. Each BRAM resource provides 36 Kb memory. In addition to BRAM, the Artix-7 also offers distributed RAMs. LUTRAM refers to the Distributed RAMs which are faster than BRAMs, however they are more limited in terms of space. FF refers to Flip Flops which are simply 1-bit memory to synchronize logic and save the logic state. BUFG is the Global clock buffers and used along with the clock generator to distribute and synchronize the clock throughout the design. IO refers to the input/output resources. For example, in order to interface the FPGA and the ADC, the FPGA input resources are used to connect the output of the ADC to the FPGA. It should be noted that in this report everything is considered including the AXI/JTAG block and any necessary debugging core. However, in the final product the debugging core can be removed. For example, by removing the AXI/JTAG block we can reduce the BRAM utilization down to 17%. Moreover, Table 6 shows that we are not using all the available resources in the selected device. As a result, in future we can migrate our design to a smaller FPGA to reduce the cost.

5.7. Summary

In this chapter, we described our test platform and explained how automation can accelerate our test procedures to reduce the test and development cost. We also presented how the FPGA and the computer can establish an efficient data communication using the AXI/JTAG block. Moreover, we have shown the FPGA utilization and concluded that it is possible to migrate to smaller FPGA. We have also explored our automation scripts in detail. In the final chapter, we conclude our thesis and discuss future works.

Chapter 6

Conclusion and Future Work

Inductive proximity sensors are used widely in aerospace industry due to their high tolerance in a harsh environment such as extreme temperature ranges. There is been interest from aircraft manufacturers to increase the accuracy of the inductive proximity sensor as well as making it more robust to temperature drift. Furthermore, the inductive proximity sensors offered in the industry have internal characteristics. Therefore, building one system to integrate multiple sensor types can reduce the cost significantly. A case study for cost reduction would be considering a commercial aircraft consisting of 50 inductive proximity sensors; each active sensor costs around $3000, then the total cost of active implementation of proxy measurement system would be $150,000. However, considering that each passive sensor costs around $700 and each proximity measurement board costs approximately $2000 which is capable to accommodate up to 10 sensors, we can calculate that the total cost of passive implementation is $45,000 which is less than a third compared to the active implementation of the measurement system.

In this thesis, we designed and implemented a proximity measurement system that is able to automatically compensate the temperature drift by separating the inductance and resistance components. Moreover, this proximity measurement system is extended to support multiple inductive proximity sensors with different internal characteristics. Such flexibility allows us to easily adapt a new inductive proximity sensor if needed in the future. Furthermore, an automated test platform is designed and implemented to reduce the test and development time as well as reducing the measurement errors. The major contributions are listed below:

1. Current ramp excitation method that automatically compensates the temperature drift.
2. Supporting up to 10 sensors with different internal characteristics in a single proximity measurement system.
3. Automating the calibration and test procedures that significantly reduces the test and development time.

Currently, the Crouzet sensor showed the best measurement accuracy among other sensors. A future work is being able to achieve the same accuracy level for other sensors. Some of the suggested future works are listed below:

1. Applying a double ramp excitation method for Goodrich and Crane sensors, this would potentially decrease the inductance fluctuations and as a result, increases the measurement accuracy.
2. Optimizing the design in order to fit it in a smaller FPGA to reduce the cost.
3. Creating a more flexible GUI to interact easily with the test platform.
4. Comparing sensor to sensor variations in one sensor model

Appendix

Table 7: Inductance Mean Error for a Crouzet Sensor at Room Temperature

Distance [mm]	L Mean Error [mH]	Distance [mm]	L Mean Error [mH]
0	0.00155133	2.6	0.001516823
0.1	0.001549881	2.7	0.00153091
0.2	0.001463734	2.8	0.001435913
0.3	0.001473557	2.9	0.001383543
0.4	0.00149934	3	0.001699809
0.5	0.001649857	3.1	0.001690304
0.6	0.001541013	3.2	0.001670688
0.7	0.001477091	3.3	0.001679778
0.8	0.001605194	3.4	0.001663424
0.9	0.001544818	3.5	0.001605779
1	0.001486946	3.6	0.001454351
1.1	0.001501616	3.7	0.001534108
1.2	0.001621476	3.8	0.001436885
1.3	0.00172779	3.9	0.00148239
1.4	0.001951754	4	0.001583222
1.5	0.001557641	4.1	0.00162496
1.6	0.001892924	4.2	0.001566477
1.7	0.001574449	4.3	0.001770969
1.8	0.001603309	4.4	0.001798982
1.9	0.001475602	4.5	0.001501929
2	0.001681399	4.6	0.001546239
2.1	0.001723528	4.7	0.001415001
2.2	0.001675112	4.8	0.001584902
2.3	0.001734853	4.9	0.001631351
2.4	0.001592957	5	0.001659259
2.5	0.001809403		

Table 8: Inductance-Distance Interpolation Table for a Crouzet Sensor

Index	Inductance	Distance	DeltaDist	ROM_MSB	ROM_LSB	ROM 32
0	22.58634615	5	0.41325721	40000	52897	9C40CEA1
1	22.70237568	4.58674279	0.323899894	36694	41459	8F56A1F3
2	22.81840521	4.262842897	0.261338323	34103	33451	853782AB
3	22.93443474	4.001504574	0.22494262	32012	28793	7D0C7079
4	23.05046427	3.776561954	0.192468376	30212	24636	7604603C
5	23.1664938	3.584093578	0.166396434	28673	21299	70015333
6	23.28252333	3.417697144	0.152588286	27342	19531	6ACE4C4B
7	23.39855286	3.265108858	0.135163351	26121	17301	66094395
8	23.51458239	3.129945508	0.120998479	25040	15488	61D03C80
9	23.63061192	3.008947029	0.109657966	24072	14036	5E0836D4
10	23.74664145	2.899289062	0.102960356	23194	13179	5A9A337B
11	23.86267098	2.796328707	0.095177764	22371	12183	57632F97
12	23.97870051	2.701150943	0.086598831	21609	11085	54692B4D
13	24.09473004	2.614552112	0.080953604	20916	10362	51B4287A
14	24.21075958	2.533598508	0.075555225	20269	9671	4F2D25C7
15	24.32678911	2.458043283	0.070960845	19664	9083	4CD0237B
16	24.44281864	2.387082438	0.067785927	19097	8677	4A9921E5
17	24.55884817	2.31929651	0.063448492	18554	8121	487A1FB9
18	24.6748777	2.255848018	0.059620018	18047	7631	467F1DCF
19	24.79090723	2.196228	0.057034148	17570	7300	44A21C84
20	24.90693676	2.139193852	0.054133057	17114	6929	42DA1B11
21	25.02296629	2.085060795	0.051002829	16680	6528	41281980
22	25.13899582	2.034057966	0.048373944	16272	6192	3F901830
23	25.25502535	1.985684022	0.046280303	15885	5924	3E0D1724
24	25.37105488	1.939403719	0.044651574	15515	5715	3C9B1653
25	25.48708441	1.894752145	0.043288892	15158	5541	3B3615A5
26	25.60311394	1.851463253	0.041652397	14812	5332	39DC14D4
27	25.71914347	1.809810856	0.039700701	14478	5082	388E13DA
28	25.835173	1.770110155	0.037881567	14161	4849	375112F1
29	25.95120253	1.732228588	0.036420702	13858	4662	36221236
30	26.06723206	1.695807886	0.035228956	13566	4509	34FE119D
31	26.18326159	1.660578929	0.034049242	13285	4358	33E51106
32	26.29929112	1.626529687	0.032836565	13012	4203	32D4106B
33	26.41532065	1.593693122	0.031624086	12750	4048	31CE0FD0
34	26.53135018	1.562069036	0.030486838	12497	3902	30D10F3E
35	26.64737971	1.531582197	0.029435137	12253	3768	2FDD0EB8
36	26.76340924	1.502147061	0.028469377	12017	3644	2EF10E3C
37	26.87943877	1.473677684	0.027591539	11789	3532	2E0D0DCC
38	26.99546831	1.446086145	0.026802241	11569	3431	2D310D67

39	27.11149784	1.419283904	0.026101157	11354	3341	2C5A0D0D
40	27.22752737	1.393182747	0.025453593	11145	3258	2B890CBA
41	27.3435569	1.367729154	0.024794639	10942	3174	2ABE0C66
42	27.45958643	1.342934514	0.024117169	10743	3087	29F70C0F
43	27.57561596	1.318817345	0.023421269	10551	2998	29370BB6
44	27.69164549	1.295396076	0.022725142	10363	2909	287B0B5D
45	27.80767502	1.272670934	0.022069525	10181	2825	27C50B09
46	27.92370455	1.250601409	0.021459947	10005	2747	27150ABB
47	28.03973408	1.229141462	0.020896408	9833	2675	26690A73
48	28.15576361	1.208245054	0.020377761	9666	2608	25C20A30
49	28.27179314	1.187867293	0.019886717	9503	2545	251F09F1
50	28.38782267	1.167980576	0.019409912	9344	2484	248009B4
51	28.5038522	1.148570664	0.018946998	9189	2425	23E50979
52	28.61988173	1.129623666	0.018497975	9037	2368	234D0940
53	28.73591126	1.111125691	0.018062808	8889	2312	22B90908
54	28.85194079	1.093062883	0.01764006	8745	2258	222908D2
55	28.96797032	1.075422824	0.017227796	8603	2205	219B089D
56	29.08399985	1.058195027	0.016825881	8466	2154	2112086A
57	29.20002938	1.041369147	0.016434314	8331	2104	208B0838
58	29.31605891	1.024934833	0.016053095	8199	2055	20070807
59	29.43208844	1.008881738	0.015682393	8071	2007	1F8707D7
60	29.54811797	0.993199345	0.015327526	7946	1962	1F0A07AA
61	29.6641475	0.977871819	0.014994774	7823	1919	1E8F077F
62	29.78017704	0.962877044	0.014684498	7703	1880	1E170758
63	29.89620657	0.948192546	0.014396698	7586	1843	1DA20733
64	30.0122361	0.933795849	0.014131373	7470	1809	1D2E0711
65	30.12826563	0.919664475	0.013888525	7357	1778	1CBD06F2
66	30.24429516	0.90577595	0.01366722	7246	1749	1C4E06D5
67	30.36032469	0.89210873	0.013452298	7137	1722	1BE106BA
68	30.47635422	0.878656432	0.013231433	7029	1694	1B75069E
69	30.59238375	0.865424999	0.013004268	6923	1665	1B0B0681
70	30.70841328	0.852420731	0.012770804	6819	1635	1AA30663
71	30.82444281	0.839649926	0.012531041	6717	1604	1A3D0644
72	30.94047234	0.827118885	0.012284979	6617	1572	19D90624
73	31.05650187	0.814833907	0.012032616	6519	1540	19770604
74	31.1725314	0.80280129	0.011775344	6422	1507	191605E3
75	31.28856093	0.791025946	0.011524698	6328	1475	18B805C3
76	31.40459046	0.779501249	0.011286392	6236	1445	185C05A5
77	31.52061999	0.768214857	0.011060468	6146	1416	18020588
78	31.63664952	0.757154389	0.010846925	6057	1388	17A9056C
79	31.75267905	0.746307464	0.010645764	5970	1363	17520553

80	31.86870858	0.7356617	0.010456985	5885	1338	16FD053A
81	31.98473811	0.725204715	0.010280587	5802	1316	16AA0524
82	32.10076764	0.714924127	0.010116571	5719	1295	1657050F
83	32.21679717	0.704807556	0.009964662	5638	1275	160604FB
84	32.3328267	0.694842895	0.009819082	5559	1257	15B704E9
85	32.44885623	0.685023813	0.009674339	5480	1238	156804D6
86	32.56488576	0.675349473	0.009530216	5403	1220	151B04C4
87	32.6809153	0.665819257	0.009386712	5327	1201	14CF04B1
88	32.79694483	0.656432545	0.009243827	5251	1183	1483049F
89	32.91297436	0.647188718	0.009101562	5178	1165	143A048D
90	33.02900389	0.638087156	0.008959916	5105	1147	13F1047B
91	33.14503342	0.62912724	0.00881889	5033	1129	13A90469
92	33.26106295	0.620308351	0.008678482	4962	1111	13620457
93	33.37709248	0.611629868	0.008538694	4893	1093	131D0445
94	33.49312201	0.603091174	0.00839964	4825	1075	12D90433
95	33.60915154	0.594691534	0.008262814	4758	1058	12960422
96	33.72518107	0.58642872	0.008129269	4691	1041	12530411
97	33.8412106	0.578299451	0.007999027	4626	1024	12120400
98	33.95724013	0.570300425	0.007872088	4562	1008	11D203F0
99	34.07326966	0.562428337	0.007748451	4499	992	119303E0
100	34.18929919	0.554679886	0.007628118	4437	976	115503D0
101	34.30532872	0.547051768	0.007511088	4376	961	111803C1
102	34.42135825	0.53954068	0.00739736	4316	947	10DC03B3
103	34.53738778	0.53214332	0.007286936	4257	933	10A103A5
104	34.65341731	0.524856384	0.007179814	4199	919	10670397
105	34.76944684	0.51767657	0.007075995	4141	906	102D038A
106	34.88547637	0.510600575	0.00697548	4085	893	0FF5037D
107	35.0015059	0.503625095	0.006878242	4029	880	0FBD0370
108	35.11753543	0.496746853	0.006783627	3974	868	0F860364
109	35.23356496	0.489963226	0.006690936	3920	856	0F500358
110	35.34959449	0.48327229	0.006600134	3866	845	0F1A034D
111	35.46562403	0.476672156	0.006511222	3813	833	0EE50341
112	35.58165356	0.470160934	0.0064242	3761	822	0EB10336
113	35.69768309	0.463736735	0.006339067	3710	811	0E7E032B
114	35.81371262	0.457397668	0.006255823	3659	801	0E4B0321
115	35.92974215	0.451141845	0.006174469	3609	790	0E190316
116	36.04577168	0.444967376	0.006095005	3560	780	0DE8030C
117	36.16180121	0.438872371	0.00601743	3511	770	0DB70302
118	36.27783074	0.432854941	0.005941745	3463	761	0D8702F9
119	36.39386027	0.426913196	0.005867949	3415	751	0D5702EF
120	36.5098898	0.421045246	0.005796043	3368	742	0D2802E6

121	36.62591933	0.415249203	0.005726027	3322	733	0CFA02DD
122	36.74194886	0.409523177	0.005657899	3276	724	0CCC02D4
123	36.85797839	0.403865277	0.005591655	3231	716	0C9F02CC
124	36.97400792	0.398273622	0.005526824	3186	707	0C7202C3
125	37.09003745	0.392746798	0.005462628	3142	699	0C4602BB
126	37.20606698	0.38728417	0.005398995	3098	691	0C1A02B3
127	37.32209651	0.381885175	0.005335926	3055	683	0BEF02AB
128	37.43812604	0.37654925	0.005273419	3012	675	0BC402A3
129	37.55415557	0.37127583	0.005211476	2970	667	0B9A029B
130	37.6701851	0.366064354	0.005150096	2929	659	0B710293
131	37.78621463	0.360914258	0.00508928	2887	651	0B47028B
132	37.90224416	0.355824978	0.005029026	2847	644	0B1F0284
133	38.01827369	0.350795952	0.004969336	2806	636	0AF6027C
134	38.13430322	0.345826616	0.004910209	2767	629	0ACF0275
135	38.25033276	0.340916407	0.004851645	2727	621	0AA7026D
136	38.36636229	0.336064763	0.004793644	2689	614	0A810266
137	38.48239182	0.331271119	0.004736206	2650	606	0A5A025E
138	38.59842135	0.326534912	0.004679332	2612	599	0A340257
139	38.71445088	0.32185558	0.004623021	2575	592	0A0F0250
140	38.83048041	0.317232559	0.004567273	2538	585	09EA0249
141	38.94650994	0.312665286	0.004512088	2501	578	09C50242
142	39.06253947	0.308153198	0.004457467	2465	571	09A1023B
143	39.178569	0.303695731	0.004403408	2430	564	097E0234
144	39.29459853	0.299292322	0.004349928	2394	557	095A022D
145	39.41062806	0.294942395	0.00429706	2360	550	09380226
146	39.52665759	0.290645335	0.00424481	2325	543	0915021F
147	39.64268712	0.286400525	0.004193179	2291	537	08F30219
148	39.75871665	0.282207346	0.004142166	2258	530	08D20212
149	39.87474618	0.278065181	0.004091771	2225	524	08B1020C
150	39.99077571	0.273973409	0.004041995	2192	517	08900205
151	40.10680524	0.269931415	0.003992837	2159	511	086F01FF
152	40.22283477	0.265938578	0.003944297	2128	505	085001F9
153	40.3388643	0.261994281	0.003896375	2096	499	083001F3
154	40.45489383	0.258097906	0.003849072	2065	493	081101ED
155	40.57092336	0.254248833	0.003802388	2034	487	07F201E7
156	40.68695289	0.250446446	0.003756321	2004	481	07D401E1
157	40.80298242	0.246690125	0.003710873	1974	475	07B601DB
158	40.91901195	0.242979252	0.003666043	1944	469	079801D5
159	41.03504149	0.239313209	0.003621831	1915	464	077B01D0
160	41.15107102	0.235691378	0.003578238	1886	458	075E01CA
161	41.26710055	0.232113139	0.003535263	1857	453	074101C5

162	41.38313008	0.228577876	0.003492907	1829	447	072501BF
163	41.49915961	0.225084969	0.003451168	1801	442	070901BA
164	41.61518914	0.221633801	0.003410048	1773	436	06ED01B4
165	41.73121867	0.218223753	0.003369547	1746	431	06D201AF
166	41.8472482	0.214854206	0.003329663	1719	426	06B701AA
167	41.96327773	0.211524543	0.003290398	1692	421	069C01A5
168	42.07930726	0.208234144	0.003251752	1666	416	068201A0
169	42.19533679	0.204982393	0.003213723	1640	411	0668019B
170	42.31136632	0.20176867	0.003176315	1614	407	064E0197
171	42.42739585	0.198592355	0.003139574	1589	402	06350192
172	42.54342538	0.195452781	0.003103557	1564	397	061C018D
173	42.65945491	0.192349224	0.003068267	1539	393	06030189
174	42.77548444	0.189280957	0.003033703	1514	388	05EA0184
175	42.89151397	0.186247253	0.002999867	1490	384	05D20180
176	43.0075435	0.183247387	0.002966756	1466	380	05BA017C
177	43.12357303	0.18028063	0.002934373	1442	376	05A20178
178	43.23960256	0.177346258	0.002902716	1419	372	058B0174
179	43.35563209	0.174443541	0.002871786	1396	368	05740170
180	43.47166162	0.171571755	0.002841583	1373	364	055D016C
181	43.58769115	0.168730173	0.002812106	1350	360	05460168
182	43.70372068	0.165918067	0.002783356	1327	356	052F0164
183	43.81975022	0.163134711	0.002755332	1305	353	05190161
184	43.93577975	0.160379379	0.002728036	1283	349	0503015D
185	44.05180928	0.157651343	0.002701466	1261	346	04ED015A
186	44.16783881	0.154949877	0.002675623	1240	342	04D80156
187	44.28386834	0.152274255	0.002650506	1218	339	04C20153
188	44.39989787	0.149623749	0.002626116	1197	336	04AD0150
189	44.5159274	0.146997633	0.002602453	1176	333	0498014D
190	44.63195693	0.14439518	0.002579516	1155	330	0483014A
191	44.74798646	0.141815664	0.002557306	1135	327	046F0147
192	44.86401599	0.139258357	0.002535823	1114	325	045A0145
193	44.98004552	0.136722534	0.002515067	1094	322	04460142
194	45.09607505	0.134207467	0.002495037	1074	319	0432013F
195	45.21210458	0.13171243	0.002475734	1054	317	041E013D
196	45.32813411	0.129236696	0.002457158	1034	315	040A013B
197	45.44416364	0.126779539	0.002439308	1014	312	03F60138
198	45.56019317	0.124340231	0.002422185	995	310	03E30136
199	45.6762227	0.121918046	0.002405788	975	308	03CF0134
200	45.79225223	0.119512258	0.002390119	956	306	03BC0132
201	45.90828176	0.117122139	0.002375176	937	304	03A90130
202	46.02431129	0.114746963	0.00236096	918	302	0396012E

203	46.14034082	0.112386004	0.00234747	899	300	0383012C
204	46.25637035	0.110038534	0.002334707	880	299	0370012B
205	46.37239988	0.107703827	0.002322671	862	297	035E0129
206	46.48842941	0.105381156	0.002311361	843	296	034B0128
207	46.60445894	0.103069795	0.002300778	825	294	03390126
208	46.72048848	0.100769016	0.002290897	806	293	03260125
209	46.83651801	0.098478119	0.002281418	788	292	03140124
210	46.95254754	0.096196701	0.002272148	770	291	03020123
211	47.06857707	0.093924553	0.002263085	751	290	02EF0122
212	47.1846066	0.091661468	0.002254227	733	289	02DD0121
213	47.30063613	0.089407241	0.002245575	715	287	02CB011F
214	47.41666566	0.087161666	0.00223713	697	286	02B9011E
215	47.53269519	0.084924536	0.00222889	679	285	02A7011D
216	47.64872472	0.082695646	0.002220856	662	284	0296011C
217	47.76475425	0.08047479	0.002213029	644	283	0284011B
218	47.88078378	0.078261761	0.002205407	626	282	0272011A
219	47.99681331	0.076056354	0.002197991	608	281	02600119
220	48.11284284	0.073858363	0.002190782	591	280	024F0118
221	48.22887237	0.071667581	0.002183778	573	280	023D0118
222	48.3449019	0.069483803	0.00217698	556	279	022C0117
223	48.46093143	0.067306823	0.002170388	538	278	021A0116
224	48.57696096	0.065136435	0.002164003	521	277	02090115
225	48.69299049	0.062972432	0.002157823	504	276	01F80114
226	48.80902002	0.060814609	0.002151849	487	275	01E70113
227	48.92504955	0.05866276	0.002146081	469	275	01D50113
228	49.04107908	0.056516678	0.00214052	452	274	01C40112
229	49.15710861	0.054376158	0.002135164	435	273	01B30111
230	49.27313814	0.052240994	0.002130014	418	273	01A20111
231	49.38916767	0.05011098	0.00212507	401	272	01910110
232	49.50519721	0.04798591	0.002120333	384	271	0180010F
233	49.62122674	0.045865578	0.002115801	367	271	016F010F
234	49.73725627	0.043749777	0.002111475	350	270	015E010E
235	49.8532858	0.041638302	0.002107355	333	270	014D010E
236	49.96931533	0.039530947	0.002103441	316	269	013C010D
237	50.08534486	0.037427506	0.002099733	299	269	012B010D
238	50.20137439	0.035327772	0.002096232	283	268	011B010C
239	50.31740392	0.033231541	0.002092936	266	268	010A010C
240	50.43343345	0.031138605	0.002089846	249	268	00F9010C
241	50.54946298	0.029048759	0.002086962	232	267	00E8010B
242	50.66549251	0.026961797	0.002084284	216	267	00D8010B
243	50.78152204	0.024877513	0.002081812	199	266	00C7010A

244	50.89755157	0.022795701	0.002079546	182	266	00B6010A
245	51.0135811	0.020716155	0.002077486	166	266	00A6010A
246	51.12961063	0.018638669	0.002075632	149	266	0095010A
247	51.24564016	0.016563037	0.002073984	133	265	00850109
248	51.36166969	0.014489052	0.002072542	116	265	00740109
249	51.47769922	0.01241651	0.002071307	99	265	00630109
250	51.59372875	0.010345203	0.002070277	83	265	00530109
251	51.70975828	0.008274927	0.002069453	66	265	00420109
252	51.82578781	0.006205474	0.002068835	50	265	00320109
253	51.94181734	0.004136639	0.002068423	33	265	00210109
254	52.05784687	0.002068217	0.002068217	17	265	00110109
255	52.1738764	0	0	0	0	00000000

Bibliography

[1] B. Marr, "That's Data Science: Airbus Puts 10,000 Sensors in Every Single Wing!," 9 April 2015. [Online]. Available: http://www.datasciencecentral.com/profiles/blogs/that-s-data-science-airbus-puts-10-000-sensors-in-every-single.

[2] H. Jiang, "Key Findings on Airplane Economic Life," March 2013. [Online]. Available: http://www.boeing.com/assets/pdf/commercial/aircraft_economic_life_whitepaper.pdf.

[3] I. M. C. T. Force, "AIRLINE MAINTENANCE COST," 2014. [Online]. Available: https://www.iata.org/whatwedo/workgroups/Documents/MCTF/AMC-Exec-Comment-FY14.pdf.

[4] G. Cros, "FY2013 Maintenance Cost Preliminary Analysis," 2014. [Online]. Available: https://www.iata.org/whatwedo/workgroups/Documents/MCC-2014-ATH/D1/1000-1030-fy13-mtce-cost-analysis-IATA.pdf.

[5] Softzone, " Proximity Sensors Theory of operation," [Online]. Available: http://www.softnoze.com/downloads/Sensor%20Basics%203.pdf.

[6] T. A. Kinney, "Proximity Sensors Compared: Inductive, Capacitive, Photoelectric, and Ultrasonic," 1 September 2001. [Online]. Available: http://machinedesign.com/sensors/proximity-sensors-compared-inductive-capacitive-photoelectric-and-ultrasonic.

[7] Softzone, "Photoelectric Sensors Theory of Operation," [Online]. Available: http://www.softnoze.com/downloads/Sensor%20Basics%204.pdf.

[8] S. S. Intelligence, "Ultrasonic sensors," [Online]. Available: https://www.sick.com/media/dox/4/64/464/Product_information_Ultrasonic_Sensors_UM30_UM18_UM12_UC30_UC12_UC4_en_IM0052464.PDF.

[9] S. S. Intelligence, "Proximity Sensors," [Online]. Available: https://www.sick.com/media/dox/8/18/618/Product_catalog_Proximity_Sensors_en_IM0048618.PDF.

[10] F. A. m. t. (. handbook. [Online]. Available: https://www.faa.gov/regulations_policies/handbooks_manuals/aircraft/amt_airframe_handbook/.

[11] P. Leons, A. Yaghoubian, G. Cowan, J. Trajkovic, Y. Nazon and S. Abdi, "On improving the range of inductive proximity sensors for avionic applications," *Sixteenth International Symposium on Quality Electronic Design,* no. IEEE, pp. 547 - 551, 2015.

[12] C. Aerospace, "Detection and Sensing - Technical Catalog," [Online]. Available: http://www.crouzet.com/crouzet_docs/documents/AEROSPACE_DETECTION_6730200EN.pdf.

[13] G. Coporation, "GOODRICH SENSORS AND INTEGRATED SYSTEMS," [Online]. Available: http://www.goodrichdeicing.com/images/stories/pdf/SISAerospaceDefenseBusinessOverview.pdf.

[14] Honeywell, "Proximity Sensors - 320FW04-5," [Online]. Available: http://sensing.honeywell.com/320FW04-5-Proximity-Sensors.

[15] Z.-B. S. †. a. T. L. †. Yi-Xin Guo, "An Analog-Digital Mixed Measurement Method of Inductive Proximity Sensor," *Sensors,* no. 1, 2016.

[16] H. Z. T. B. a. G. B. B. George, "A Combined Inductive–Capacitive Proximity Sensor for Seat Occupancy Detection," *IEEE Transactions on Instrumentation and Measurement,* vol. 59, no. 5, pp. 1463-1470, 2010.

[17] M. R. Nabavi and S. Nihtianov, "Stability considerations in a new interface circuit for inductive position sensors," *Electronics, Circuits, and Systems,* pp. 932-935, 2009.

[18] N. Mehta, "Xilinx 7 Series FPGAs," [Online]. Available: http://www.xilinx.com/support/documentation/white_papers/wp377_7Series_Embed_Mem_Advantages.pdf.

[19] Xilinx, "7 Series DSP48E1 Slice," [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug479_7Series_DSP48E1.pdf.

[20] C. A. &. Electronics, "MODEL 8-642," [Online]. Available: http://www.craneae.com/Products/Sensing/Downloads/8-642.pdf.

[21] Newport, "Miniature linear stage, DC motor," [Online]. Available: https://www.newport.com/p/MFA-CC.

[22] Xilinx, "Artix®-7 FPGAs," [Online]. Available: http://www.xilinx.com/support/documentation/selection-guides/artix7-product-table.pdf.

[23] N. Corporation, "Miniature linear stage, DC motor," [Online]. Available: https://www.newport.com/p/MFA-CC.

[24] M. Jagiella, S. Fericean and A. Dorneich, "Progress and recent realizations of miniaturized inductive proximity sensors for automation," *IEEE J. Sens,* p. 1734–1741, 2006.

[25] Xilinx, "Artix-7 Product Advantage," [Online]. Available: http://www.xilinx.com/products/silicon-devices/fpga/artix-7.html.

[26] Xilinx, "7 Series FPGAs Memory Resources," [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug473_7Series_Memory_Resources.pdf.

[27] A. Yang, "Using SPI Flash with 7 Series FPGAs," 2013 January 31. [Online]. Available: http://www.xilinx.com/support/documentation/application_notes/xapp586-spi-flash.pdf.

[28] Micron, "Micron Serial NOR Flash Memory," [Online]. Available: https://www.micron.com/~/media/documents/.../n25q_128mb_1_8v_65nm.pdf.

[29] Xilinx, "JTAG to AXI Master v1.0," [Online]. Available: http://www.xilinx.com/support/documentation/ip_documentation/jtag_axi/v1_0/pg174-jtag-axi.pdf.

[30] pySerial, "pySerial," [Online]. Available: https://pythonhosted.org/pyserial/.