



Project seminar

DEVELOPMENT OF AN ELECTRICAL MEASUREMENT SETUP FOR THE CHARACTERISATION OF QUARTZ TUNING FORKS

AUTHOR : SATHYAJITH SHANTHKUMAR (65442)

PROFESSOR : PROF. DR.-ING. STEFFEN STREHLE

SUPERVISOR : BÜŞRA ÖYKÜ ALAN, JOHANNES TORBEN BRUNO KRECH

DATE : 31-05-2023

Declaration

I hereby declare that this report was created autonomously without using other than the stated references. All parts which are cited directly or indirectly are marked as such. This report has not been used in the same or similar forms in parts or total in other examinations.

I am aware that my work may be subject to a plagiarism check.

(Sathyajith Shanthkumar)

Ilmenau, 31.05.2024

Contents

Motivation	1
1 Introduction	2
2 State of the art	3
2.1 AFM Techniques: Optical and Electrical	3
2.2 Static and dynamic Atomic Force Microscopy (AFM)	4
2.3 Quartz Tuning Fork (QTF) in AFM	5
2.4 Project objective and methodology	7
3 System Design and Implementation	9
3.1 Circuit design: Hardware	10
3.2 Software Circuit Design	11
3.2.1 Field Programmable Gate Array (FPGA) Virtual Instrument (VI): Working	14
3.2.2 Host VI	17
4 Frequency Sweep Results	20
4.1 Presettings	20
4.2 QTF in vacuum	22
4.3 QTF in Atmosphere	24
4.4 QTF with wire tip attached	25
5 Conclusion	28
5.1 Discussion of results	28
5.2 Scope for improvement	30
5.3 Outlook	31
Bibliography	33

List of Figures

2.1	AFM using optical measurement technique: An oscillator generates the signal used to actuate the QTF mechanically. The amplitude of oscillation of the QTF is measured through a laser Doppler vibrometer and the results are displayed on the Personal Computer (PC)	4
2.2	Spring Mass equivalent of a QTF: M is the equivalent mass of the tuning fork,k is the stiffness and h is the friction between the tuning fork and the surrounding particles.	6
2.3	Dimensions of the QTF represented using a general tuning fork model	6
2.4	Electrical Model of a QTF: The resistor (R), inductor (L) and capacitor (C_0) in series and the capacitance due to the quartz substrate given by (C_1) together represent the electrical equivalent of the QTF	7
3.1	VI to FPGA Board Data transfer: The VI written by the user is converted to a VHSIC Hardware Description Language (VHDL) description by the FPGA compiler module. Later, the Xilinx compiler converts the VHDL description into a bitfile to configure the FPGA to ensure efficient resource utilization.	10
3.2	Hardware Circuit: The Host PC directs the FPGA board to send and receive an analog signal from the QTF. Data containing information about the analog signal is sent back to the PC by the FPGA board. The actuation signal from the FPGA is passed through a voltage divider, which reduces the voltage by a factor of 10 before being input to the tuning fork. The signal from the tuning fork is a small current, which is converted to voltage and later amplified by the Op-Amp before returning to the FPGA.	11

List of Figures

3.3	FPGA VI Front Panel: User can enter the frequency value used for the tuning fork actuation and the count-ticks (these values are obtained from the HOST VI when it runs as well). This is used to control the loop time of the signal-receiving loop in case of a data overflow. Time out lights up in case of a data overflow and the Root Mean Squared (RMS) indicator gives the RMS value of the oscillation amplitude. The waveform chart 2 indicates the tuning fork oscillation amplitude and the RMS Waveform graph indicates the RMS of the output voltage of the QTF.	12
3.4	FPGA VI back panel: The Loop on top is used for analog signal generation and the bottom loop performs 2 functions; It receives the output signals from the tuning fork, calculates the RMS value and sends it to the host PC via the First In First Out (FIFO)	13
3.5	Step-by-step Guide: The flowchart highlights the key stages in developing an efficient FPGA VI on LabVIEW. The first 4 stages of the flowchart are done on the Host PC VI part of the project, while the next four stages of the VI are done on the FPGA VI	14
3.6	Sinewave generator configuration: The parameters of the sinewave generator are displayed here. By default, the frequency is set to 100KHz. The analog signal amplitude is set to 5 volts, which is reduced to 0.5 volts by the voltage divider circuit before reaching the QTF. The Look-up table size and amplitude resolution are set and the frequency value is given as input to the generator in the form of period/tick. This value is obtained by dividing the frequency of the actuating analog signal by 40000000 Hz, which is the top-level clock of the FPGA.	15
3.7	FIFO: Data structure of the FIFO. It indicates the data entry (write) and data retrieval (read) action of the FIFO. It follows a queue system, thereby ensuring efficient data management. [1]	16
3.8	Host VI Block Diagram: Loop 1 calculates the frequency values required for the frequency sweep, Loop 2 interacts with the FPGA (sends and receives data) and Loop 3 writes the recorded data into an Excel File. Loop 4 makes sure the Loops 1,2 and 3 run parallelly.	17

List of Figures

3.9	Block diagram showing the host VI Loop 1 data flow: The Loop takes the start, End frequency and the step value entered by the user and generates frequency values through the range specified for each loop iteration. The frequency values are sent to Loop 2 which is later sent to the FPGA for tuning fork actuation and also to Loop 3 to be stored in the Excel file	18
3.10	Host VI Loop 2 Data Flow: This loop controls the parameters of the FPGA VI. It receives and sends data simultaneously to and from the FPGA. The frequency values from Loop 1 are transferred to the FPGA via this loop and also the RMS Value of the oscillation Amplitude from the FPGA is sent to Loop 3. In case of any data overflow, "time out" of the is also indicated in this Loop	19
4.1	Host VI Front Panel: The starting frequency, ending frequency and step value must be entered by the user before running the VI. The timeout, elements remaining and actual depth are parameters of the FIFO, used to control the data flow. If the timeout light is on or the number of elements remaining equals the actual depth, this indicates a data overflow and hence a data loss from the FIFO. Loop time provides the time taken to complete one iteration of the data processing loop. The 'Store Data' button is a switch used to control data storage in the Excel file. The RMS value of the signal from the tuning fork is also displayed.	21
4.2	QTF in Vacuum/capped: The RMS amplitude reaches a maximum at a frequency of 32762 Hz and reaches a peak of 3.86 Volts. This indicates the first Eigenmode of the QTF	23
4.3	QTF in Atmosphere: The RMS amplitude reaches a maximum of 2.3 Volts at the frequency of 32758 Hz. This indicated the first Eigen mode of this variation of the QTF	24
4.4	Tungsten tip is attached to one of the prongs of the QTF using silver epoxy . . .	26
4.5	QTF in atmosphere with tip attached: The maximum RMS Oscillation values of the tuning fork in this variation at its Eigen mode is 1.26 but due to the coupling of the fork as a result of the additional mass, there is a gradual increase in the frequency till 55000 Hz	26

List of Figures

5.1	Actuation signal images as seen on the oscilloscope: With the increase in frequency, the signal integrity decreased. The frequency value keeps changing even though there is no instruction from the FPGA board to do so. Signals 2 and 3 show distorted signals at 50,000 Hz and 90,000 Hz. This condition worsens when the frequency value is above 100 kHz.	29
5.2	Producer Consumer Loop design architecture is used to transfer data between loop 2 and loop 3 of host VI. The usage of Local variables is avoided to store the data in the Excel file in this design. Hence, there is a buffer available to avoid data loss due to desynchronization between the different loops. Consequently, it would be possible to store data even with Loops running at different rates	31

Acronyms

AFM Atomic Force Microscopy

QTF Quartz Tuning Fork

AM AFM Amplitude Modulated Atomic Force Microscopy

FM AFM Frequency Modulated Atomic Force Microscopy

FIFO First In First Out

PC Personal Computer

FPGA Field Programmable Gate Array

DMA Direct Memory Access

VI Virtual Instrument

RMS Root Mean Squared

VHDL VHSIC Hardware Description Language

NI National Instruments

VHSIC Very High Speed Integrated Circuit

Abstract

Since its inception in 1986, AFM has found its application in various fields and plays an important role in understanding material properties at the micro- and nanoscale. Two main sensors are used in AFM, namely the cantilever beam and QTF. Hence, it is important to characterize the sensors to understand their working, the forces involved, and their interaction with the sample surface. A large number of research articles have been published concerning this topic [2] [3] [4].

Characteristics of the sensor such as stiffness, spring constant, and quality factor have to be analyzed and compared in different configurations of the tuning fork and chosen according to the application. One such important characteristic is the first resonant frequency, also called the first Eigenmode of the tuning fork. It is one of the most common modes of oscillation used for both cantilever sensor and QTF sensor applications. The amplitude of oscillation is at a maximum in this mode and the frequency is also called the natural frequency of the sensor.

In this project, LabVIEW is used as the software platform to design a circuit to determine the resonant frequency of different variations of the QTF. The National Instruments (NI) 7856 R-EOM FPGA board combined with an external electrical circuit is used as the hardware for this application. The LabVIEW program establishes a connection between the host PC and the FPGA. The FPGA board can be remotely controlled from the host PC. An analog signal is generated by the FPGA board. This signal serves as the actuation input to the tuning fork and the oscillation amplitude of the fork is recorded in the form of voltage. Hence, both the actuation and detection happen through the FPGA board. The user enters a frequency range and the step value before running the program. The value of the actuation frequency and the respective RMS value of the amplitude of oscillation of the tuning fork is stored in an Excel file. Further improvement can be done by using different architectures of the LabVIEW circuit and by using different hardware components. This will be discussed in the results section of the report.

Chapter 1

Introduction

Since its inception in 1986 [5], AFM has played a crucial role in understanding the properties of materials in the micro and nanoscale. A wide range of research has been done where AFM finds its application in various fields such as fabrication, bioscience, food science and textile research. [6–8]. AFM in simple terms uses an elastic probe in combination with a sensor setup, capable of measuring the probe’s deformation. The tip of the probe is then brought close to, or into contact with the surface of the sample. By monitoring the probe’s deformation, the interaction between its tip and the surface of the sample can be detected. This interaction is used to analyze the characteristics of the sample topology. Cantilever beams and QTF are the two main sensors used in AFM. In recent times, researchers have been using QTF as an alternative to cantilever based sensors. For certain use cases, additional masses are also attached to the QTF. This alters certain properties of the tuning fork. Hence, it becomes important to study their characteristics to increase their performance and allow for proper usage in AFM and related applications. One such property is the resonance frequency. For modified QTFs, it often deviates from the values that are provided by the manufacturer, which leads to the necessity of determining the actual resonant frequencies. Therefore, an electrical setup and a method for determining the actual resonant frequencies of modified QTF are developed and validated in this project.

Chapter 2

State of the art

To improve the performance and widen the range of possible use cases of the QTF, researchers have modified them by adding tips or by fixing one of the prongs. When a tip is attached to one of the prongs of the tuning fork, an additional mass is added to it. This increases the complexity of the measurement as well as there is a higher chance of asymmetrical oscillation of the QTF because of the imbalances caused by the additional mass. To avoid this, researchers make modifications to the tuning fork. One such example is the qPlus QTF sensor. In this variation, one of the prongs of the tuning forks is fixed and the other prong is allowed to oscillate. This causes the QTF to behave like a cantilever beam and avoid asymmetrical oscillation caused by the additional mass [9]. This deviation of resonant frequency from the values that are provided by the manufacturer makes it necessary to determine the actual resonant frequencies to ensure their proper usage. For this purpose, a method for determining the actual resonant frequencies of modified QTF is developed and validated in this project.

2.1 AFM Techniques: Optical and Electrical

The measurement of deformation in AFM is done using either the optical method or the electrical method. The majority of the commercially available AFM devices use an optical detector that detects a laser reflecting off of the top of the probe to study the surface characteristics [2] [10]. A simple block diagram showing the working of the optical method is shown in the Figure 2.1. Optical measurement of the oscillating QTF is complex and has its drawbacks as discussed in [11]. In order to reduce complexity and allow for quick and automated characterization of the modified QTFs, the approach of electrical measurement is chosen for determining their resonant frequency.

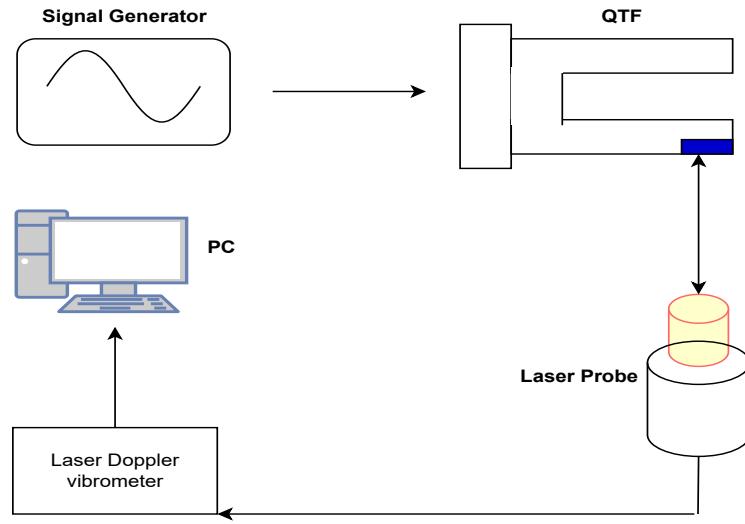


Figure 2.1: AFM using optical measurement technique: An oscillator generates the signal used to actuate the QTF mechanically. The amplitude of oscillation of the QTF is measured through a laser Doppler vibrometer and the results are displayed on the PC.

The two main forces involved in the interaction between the sensor tip and the sample surface are the Coulomb and Van der Waals forces. These forces are opposite to each other i.e. Coulomb forces are repulsive. They arise from the electrostatic repulsion between the tip and the sample. Van der Waals forces are attractive forces due to the fluctuating dipole. Their range is up to 10 nm or more. The stiffness of the cantilever/tuning fork plays an important role in the overall sensitivity. If the stiffness is too high compared to the force between the tip and the sample, the bent beam cannot detect the small surface lattices [12] [13]. If the stiffness is too low, the beam does not move back to the initial position and the attractive forces become too dominant. Usually, a spring constant comparable to the force gradient between the tip and the sample is chosen and it is often in a value range around 2000 N/m [13].

The most common mode used for AFM as well as other applications is the in-plane symmetrical oscillation mode which, for a QTF occurs at 32768 Hz. The experiment conducted confirms this along with analyzing the resonance frequency shift that occurs when an additional mass is attached to one of the prongs. There are a couple of different approaches to solve this problem that are discussed in the report.

2.2 Static and dynamic AFM

AFM is broadly categorised into static and dynamic AFM. In static mode, the tip is in continuous contact with the sample and the deflection of the beam is used to measure the points

required to generate a 3D image of the surface of the sample. This method has its drawbacks. Since the tip is in continuous contact with the sample, there is a high possibility of the tip getting damaged or worn out.

In dynamic mode, the probe is subjected to oscillation and it is moved over the surface of the substrate. Dynamic mode can be divided into amplitude modulated AFM and frequency modulated AFM [14]. In Amplitude Modulated Atomic Force Microscopy (AM AFM), the probe is subjected to excitation near its resonant frequency, where it oscillates with the highest amplitude and as it is brought close to the surface, the change in amplitude is measured. This is the main parameter used in mapping the sample topology. Frequency Modulated Atomic Force Microscopy (FM AFM) uses the frequency shift as a feedback source to map the sample surface. In FM AFM, the change in amplitude is ignored and only the change in frequency is measured. AM AFM and FM AFM can be used in tapping mode, an intermittent mode between contact and non-contact mode where the tip is continuously tapped over the sample surface.

2.3 QTF in AFM

In AFM, particularly dynamic AFM tuning forks have been widely used as an alternative to cantilever beam sensors. The tuning forks are made of quartz due to the piezoelectric characteristic. QTF have a high quality factor indicating a low energy loss, making them stable and precise oscillators. Unlike a cantilever beam, a QTF does not need an external mechanical actuator. Due to the piezoelectric property, a voltage can be applied directly to the QTF to cause a mechanical oscillation. It requires an actuation voltage of 0.5 to 1 Volt [3]. The alternating actuation voltage is applied to the contacts of the QTF and the resulting current is measured. A model of the QTF that is used for the project is shown in Figure 2.3.

It is important to understand the dynamic properties of the QTF to understand the interaction with the sample material surface. These properties include resonance frequency, quality factor and peak amplitude. For simplicity, a QTF can be modeled as a damped mechanical oscillator as seen in Figure 2.2.

To represent it as a spring-mass system, the behavior of the prongs and the base are collectively taken into account. The stiffness (k), damping due to resistance (h) and inertia due to mass (M) are the three main parameters. Damping is very low in a vacuum whereas it increases in the air due to resistance between the air particles and the prongs of the tuning fork [3]. Similar to a mechanical model, QTF can be represented as an electrical model too.

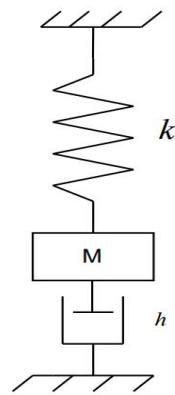


Figure 2.2: Spring Mass equivalent of a QTF: M is the equivalent mass of the tuning fork, k is the stiffness and h is the friction between the tuning fork and the surrounding particles.

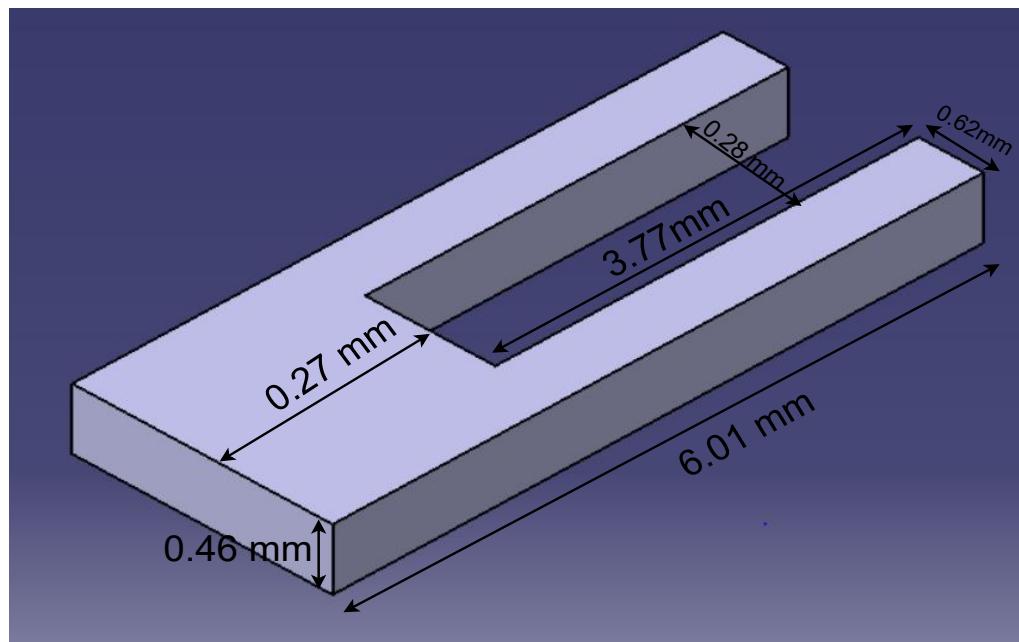


Figure 2.3: Dimensions of the QTF represented using a general tuning fork model

In this case, a resistor (R), capacitor (C_0) and inductor (L) represent the tuning fork. The capacitor (C_1) represents the capacitance due to the quartz substrate. Each prong consists of metal plates separated by the quartz substrate which gives rise to micro capacitance and these run parallel to the $R-L-C_0$ series.

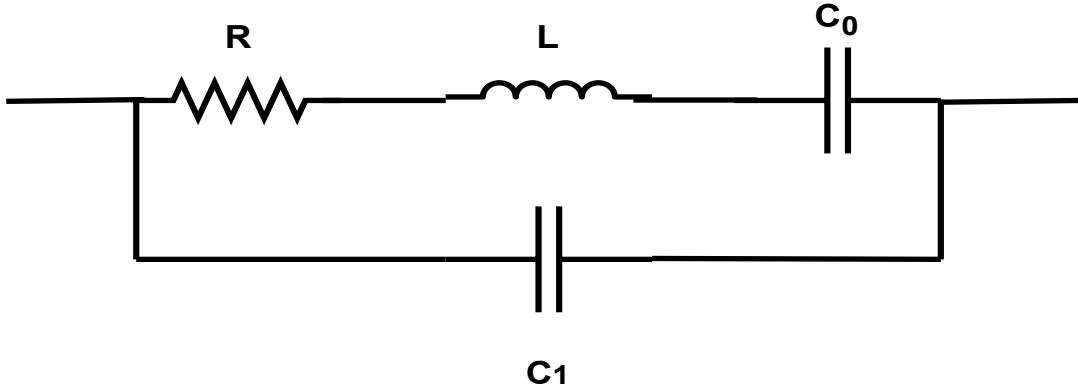


Figure 2.4: Electrical Model of a QTF: The resistor (R), inductor (L) and capacitor (C_0) in series and the capacitance due to the quartz substrate given by (C_1) together represent the electrical equivalent of the QTF

In AM AFM, the amplitude change of the tuning fork is due to damping which increases as the probe tip gets closer to the surface. The oscillating amplitude of the tuning fork is maximum at its resonant frequency. The resonant frequency depends on the geometry and the material of the tuning fork. For a tuning fork, the resonant frequency [3] is given by the formula

$$\omega = \frac{1.76a}{\ell^2} \sqrt{\frac{E}{\rho}}, \quad (2.1)$$

where ℓ is the length of the prong, a is the prong thickness, E is the Youngs modulus and ρ is the density of quartz. The QTF has multiple modes of vibration called Eigen modes [2]. When a tuning fork is allowed to oscillate naturally, it oscillates at the resonant frequency and this state is called the fundamental mode of vibration. Using FEM Simulations, the frequency values of different eigenmodes have been found [10]. There is a spike in the oscillation amplitude at these frequencies and depending on the application, one of these modes is chosen as the frequency of operation.

2.4 Project objective and methodology

- The project involves developing an electrical setup utilizing an FPGA board.

2 State of the art

- Designing a program using LabVIEW software that generates a range of frequencies and records the oscillation amplitude for three variations of QTF.
- Using Matlab to identify the resonant frequency, a graph of RMS amplitude against the respective frequencies is plotted.

This is done with three different iterations of the tuning forks, namely, a vacuum-capped QTF, a freely oscillating QTF and a QTF with a tungsten tip attached to one of the prongs. The shift in the first resonant frequency and the change in amplitude of oscillation are analyzed for these configurations. There is ongoing research to find the applications of modified quartz tuning forks in fields such as AFM and material deposition and this project setup can be used to analyze the resonant frequency and helps to automate the process for a fixed frequency range. The electrical components used in the circuit have some limitations which are discussed further in the report.

Chapter 3

System Design and Implementation

To Explain the design and implementation, the circuit is divided into two parts: Software and Hardware. All the hardware components used for actuation and measurement are shown below.

- **Host PC:** Windows OS, LabVIEW Software including the LabVIEW FPGA Module, XILINX compiler
- **FPGA Board:** NI USB 7856R OEM
- **Operational Amplifier:** Amplifie Voltcraft VSP 2405 HE
- **Oscilloscope:** Keysight Infiniivision DSOX 2014A
- **Resistors:** Axial Lead resistor used for Voltage divider circuit

The flowchart in the Figure 3.1 gives a better understanding of the connection between the LabVIEW VI and FPGA board. The VI created is converted to VHDL code. Then, the Xilinx compiler tool is invoked and the VHDL code is optimized to a hardware circuit realization of the VI. Here, timing constraints are used for efficient use of the FPGA board. From the tool, a bitfile is generated which is then transferred to the FPGA chip to configure it.

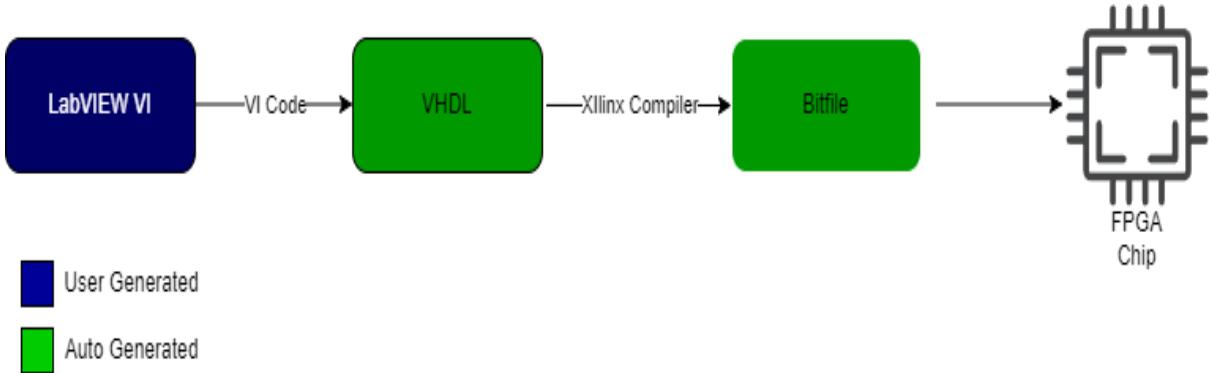


Figure 3.1: VI to FPGA Board Data transfer: The VI written by the user is converted to a VHDL description by the FPGA compiler module. Later, the Xilinx compiler converts the VHDL description into a bitfile to configure the FPGA to ensure efficient resource utilization.

3.1 Circuit design: Hardware

First, the LabVIEW software along with all the required modules to successfully compile and make real-time measurements using the FPGA Board are installed. Then, a connection is established between the FPGA and HOST PC using a USB cable. The FPGA Board analog output is connected to the tuning fork and the output of the tuning fork is connected back to the FPGA board. The output signal of the tuning fork is amplified by the operational amplifier before feeding it back to the FPGA. The FPGA board does not have an operating system of its own, hence it is necessary to load the circuit specification into the board. A block diagram of the hardware can be seen in Figure 3.2

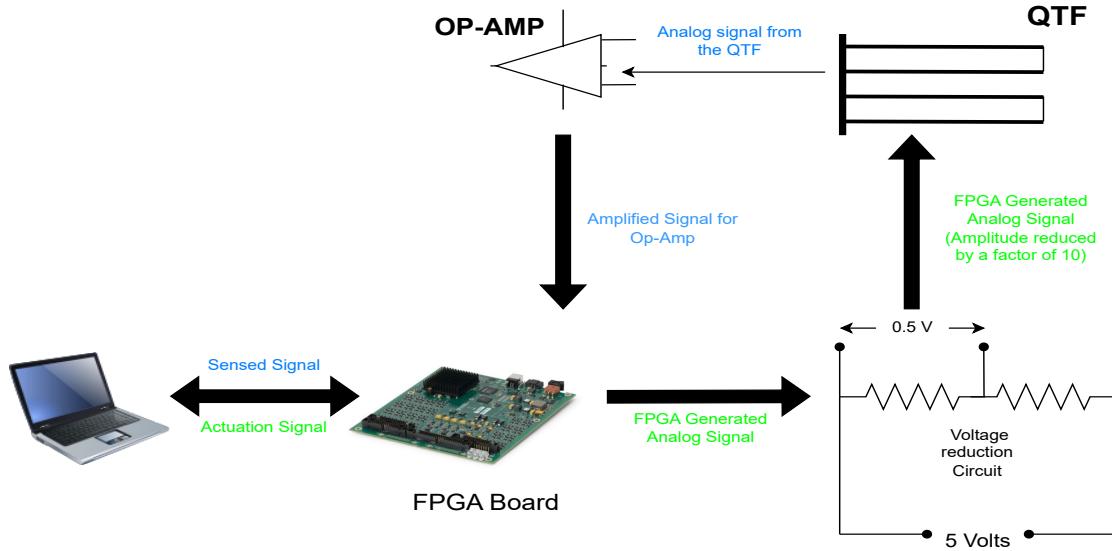


Figure 3.2: Hardware Circuit: The Host PC directs the FPGA board to send and receive an analog signal from the QTF. Data containing information about the analog signal is sent back to the PC by the FPGA board. The actuation signal from the FPGA is passed through a voltage divider, which reduces the voltage by a factor of 10 before being input to the tuning fork. The signal from the tuning fork is a small current, which is converted to voltage and later amplified by the Op-Amp before returning to the FPGA.

A signal is generated in the Host PC using the LabVIEW software. This signal directs the FPGA Board to generate analog signals. These analog signals are fed into a voltage divider circuit where they get reduced by a factor of 10. This reduced signal is used as the input to the tuning fork. The oscillation amplitude of the tuning fork is very low. Hence, an Operational Amplifier is used to amplify this signal from the tuning fork before sending it back to the FPGA Board. Data containing information about this signal is then sent back to the Host PC. The information is displayed on the host VI front panel and stored in a CSV file.

3.2 Software Circuit Design

LabVIEW is a graphical interface programming language. Here, the program files are called VI. It stands for Virtual Interface. A VI can be further divided into a front and back panel.

The **front panel** is what the users see when they open a VI (Figure 3.3). The data generated by the VI is displayed here. It also contains elements such as controls and indicators. Control elements include knobs, switches, etc. Indicators are LEDs, graphs, etc. The **back panel** is also called a block diagram. It consists of the source code of the VI. The controls on the front panel are seen here as blocks that act as terminals to the logic used. The back panel

3 System Design and Implementation

of the FPGA VI is shown in the Figure 3.4. Thus, the user interacts with the front panel to control the VI and the back panel is the functionality block. The software circuit is divided into 2 parts, the host VI and the FPGA VI. The general development procedure followed to create a VI is shown in the Figure 3.5 [15].

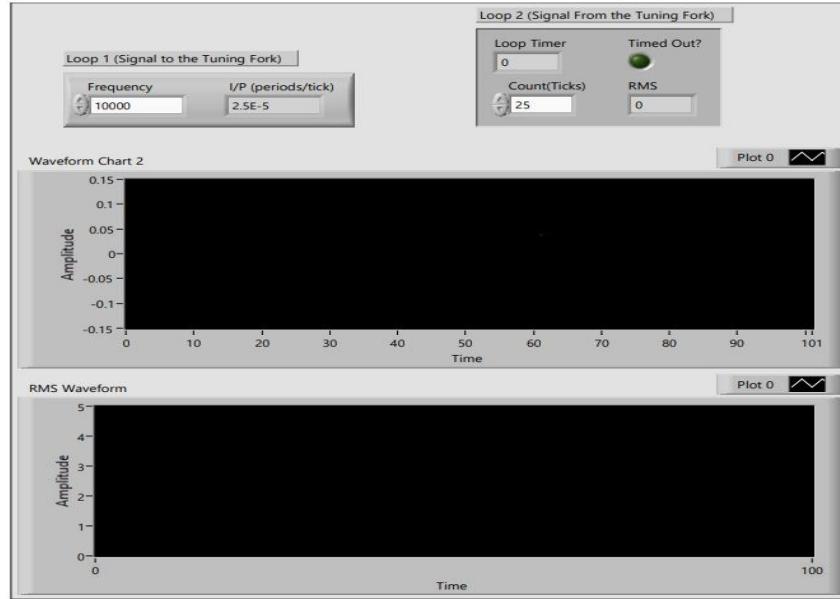


Figure 3.3: FPGA VI Front Panel: User can enter the frequency value used for the tuning fork actuation and the count-ticks (these values are obtained from the HOST VI when it runs as well). This is used to control the loop time of the signal-receiving loop in case of a data overflow. Time out lights up in case of a data overflow and the RMS indicator gives the RMS value of the oscillation amplitude. The waveform chart 2 indicates the tuning fork oscillation amplitude and the RMS Waveform graph indicates the RMS of the output voltage of the QTF.

3 System Design and Implementation

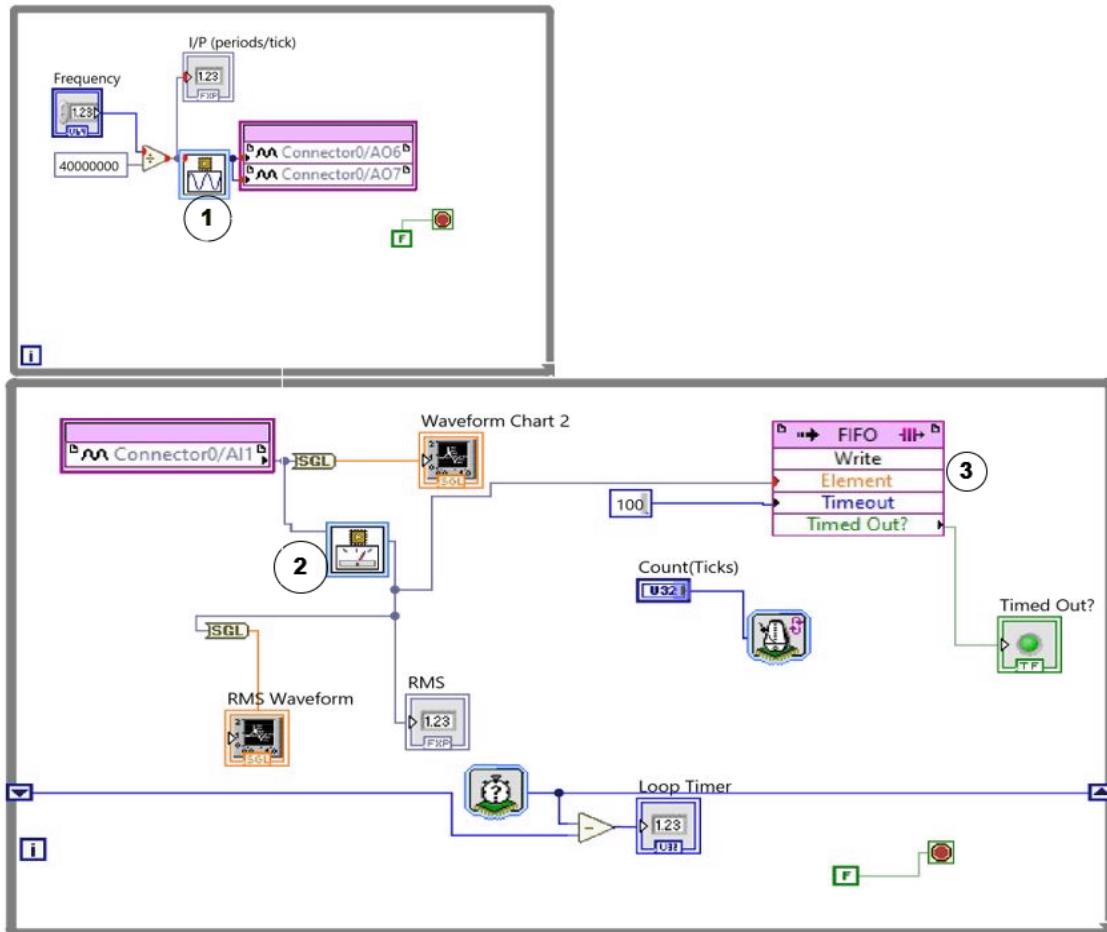


Figure 3.4: FPGA VI back panel: The Loop on top is used for analog signal generation and the bottom loop performs 2 functions; It receives the output signals from the tuning fork, calculates the RMS value and sends it to the host PC via the FIFO

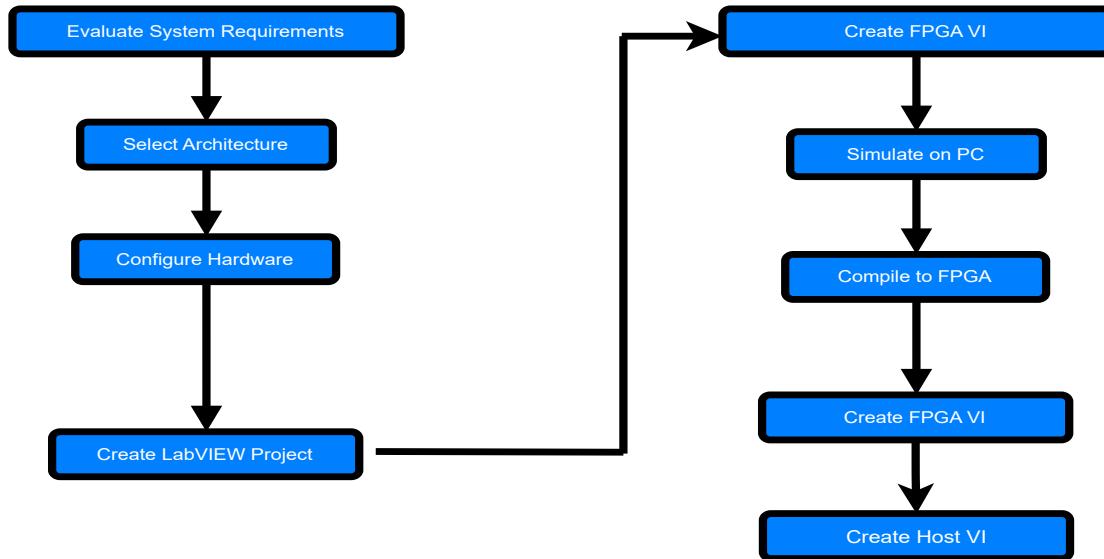


Figure 3.5: Step-by-step Guide: The flowchart highlights the key stages in developing an efficient FPGA VI on LabVIEW. The first 4 stages of the flowchart are done on the Host PC VI part of the project, while the next four stages of the VI are done on the FPGA VI

The steps involved in setting up the project environment for the VI are as follows:

1. Power up the FPGA board and configure the hardware from the NI project environment on the host PC, thereby ensuring a connection between the FPGA Board and the host.
2. There are two VIs that have to be created:
 - FPGA VI to generate and collect analog signals from the tuning fork
 - Host PC VI to interact with the FPGA VI
3. Design a circuit to store the data for further processing

3.2.1 FPGA VI: Working

The FPGA VI back panel is divided into two loops, one to generate the analog signals and one to record the signals from the tuning fork.

Loop 1 : A sinewave generator (1,Figure 3.4) is used to generate the required analog signal to excite the tuning fork. The configuration of the Sine wave generator is also shown in Figure 3.6. The three main parameters that need to be set in order to get a high-resolution sinusoidal signal are the look-up table size, the amplitude indicator and the bit count.

3 System Design and Implementation

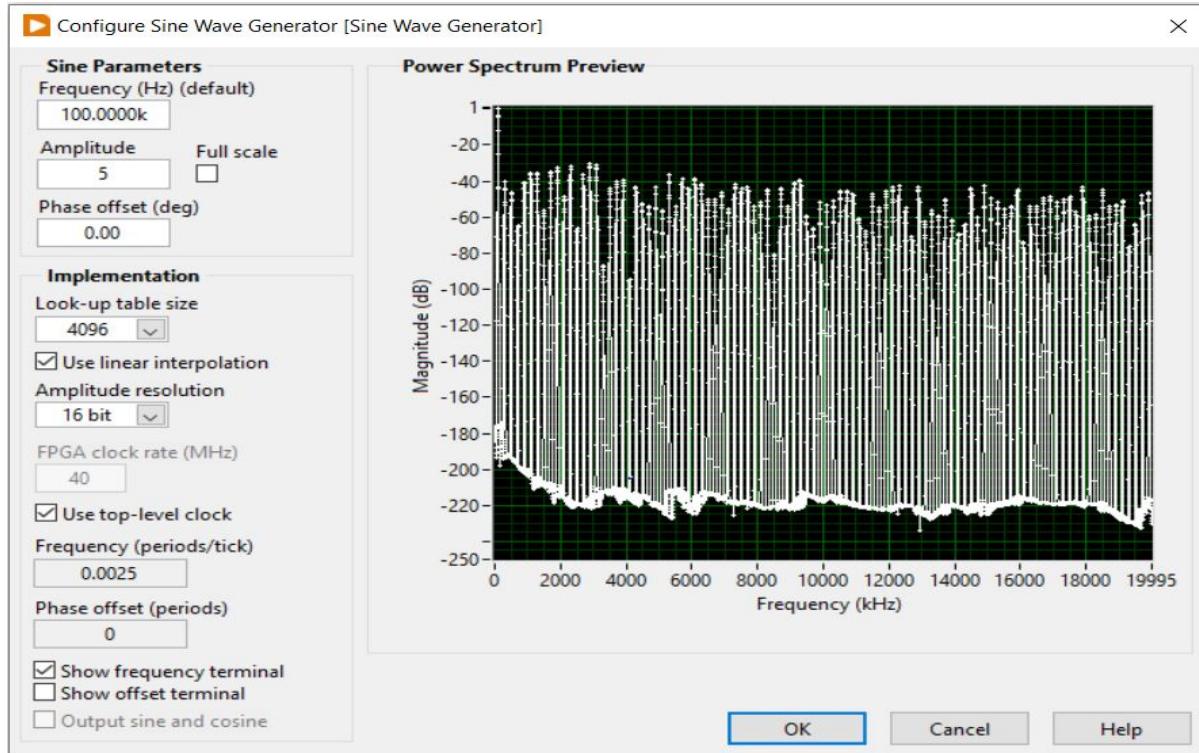


Figure 3.6: Sinewave generator configuration: The parameters of the sinewave generator are displayed here. By default, the frequency is set to 100KHz. The analog signal amplitude is set to 5 volts, which is reduced to 0.5 volts by the voltage divider circuit before reaching the QTF. The Look-up table size and amplitude resolution are set and the frequency value is given as input to the generator in the form of period/tick. This value is obtained by dividing the frequency of the actuating analog signal by 40000000 Hz, which is the top-level clock of the FPGA.

3 System Design and Implementation

As seen in Figure 3.6, the default value of the frequency is 10000Hz. This frequency is controlled from the front panel. Here, the sinewave is generated point by point and the look-up table size indicates the number of points that are stored in the block memory of the FPGA. The amplitude of the sinewave is set to 5 Volts. This sinewave is sent to the analog output 6 and 7 of the FPGA. Output 6 is connected to the voltage divider circuit and output 7 is connected to the oscilloscope to analyze the generated signal.

Loop 2 : The second loop of the FPGA VI consists of the analog input slot which receives signal from the tuning fork, an RMS value generator and a loop timer to measure the time taken to complete one iteration of the loop. RMS value is used (2, Figure 3.4) to get a generalized mean of the analog signal from the QTF and using the "hanning window" setting from the RMS function, noise can be reduced. RMS value of the voltage is given by

$$V_{\text{RMS}} = \frac{V_{\text{PK}}}{\sqrt{2}} \quad (3.1)$$

V_{pk} is the peak voltage of the sinewave. This signal is displayed on the front panel as RMS Waveform. Waveform chart 2 displays the analog signal from the tuning fork directly (without RMS conversion). Next, the measured values of the analog signals after conversion to RMS values are sent to the host PC using a data transferring tool called FIFO. This is a data handling tool that sends the data points of the RMS signal received from the QTF in the same order. The Figure 3.7 is a graphical representation of FIFO.

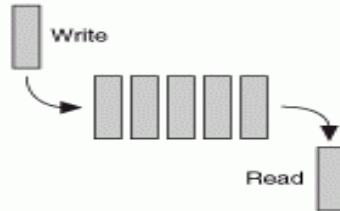


Figure 3.7: FIFO: Data structure of the FIFO. It indicates the data entry (write) and data retrieval (read) action of the FIFO. It follows a queue system, thereby ensuring efficient data management. [1]

Among the different types of FIFO available in LabVIEW, Direct Memory Access (DMA) FIFO is used. The primary function of DMA FIFO is data transfer between a host processor and an FPGA. DMA FIFO uses resources from both the host processor and the FPGA processor to reduce the FPGA memory usage and increase efficiency. When a FIFO is added to a project, the FIFO size (Number of elements), data type and the type of FIFO (target to host or host to target) have to be specified.

3 System Design and Implementation

3.2.2 Host VI

The host VI is the main part of the LabVIEW circuit. It receives data from the FPGA VI and the data is stored in the host PC memory. The back panel of the host VI is shown in Figure 3.8.

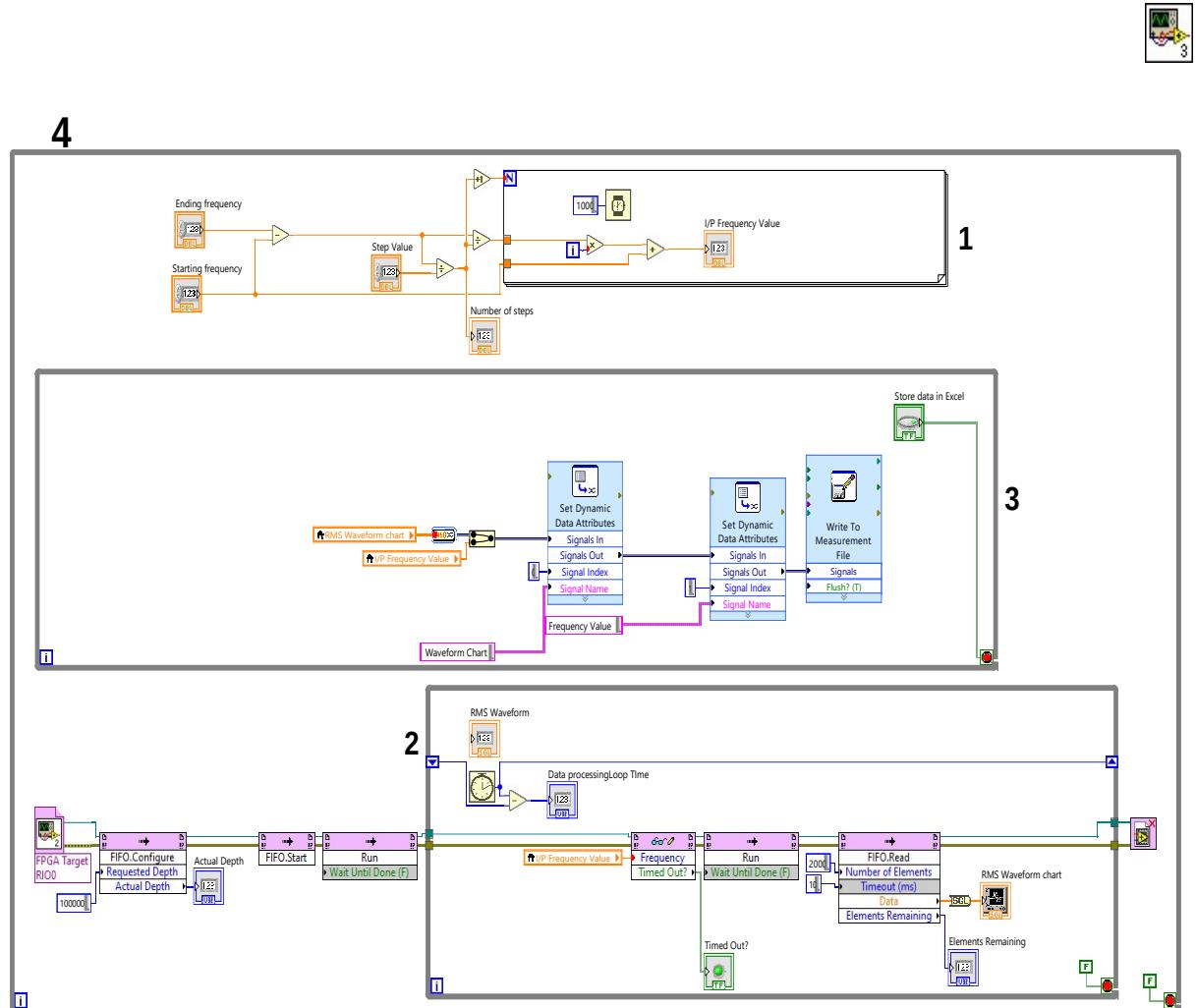


Figure 3.8: Host VI Block Diagram: Loop 1 calculates the frequency values required for the frequency sweep, Loop 2 interacts with the FPGA (sends and receives data) and Loop 3 writes the recorded data into an Excel File. Loop 4 makes sure the Loops 1,2 and 3 run parallelly.

The host VI block diagram is divided into 4 loops as labeled in Figure 3.8 and the function of each loop is listed below.

- 1. Loop 1:** Loop 1 of the host VI is used to generate individual frequency values required for the excitation of the tuning fork. The starting frequency f_{start} , the ending frequency f_{end} and the step value (S) have to be entered by the user. The frequency value f_i that is

transferred to the FPGA is calculated for each iteration I of the loop. The data generation and transfer from loop 1 to both loop 2 and 3 can be seen in Figure 3.9. Local variables are used to transfer the data between the loops. If the calculated value of n is 10, this is fed as input to the loop iteration counter and the loop runs 10 times. The frequency calculated in Loop 1 is shown below:

- a) The number of steps is calculated:

$$n = \frac{f_{\text{end}} - f_{\text{start}}}{S}$$

- b) Frequency value is calculated:

$$f_i = f_{\text{start}} + I \cdot S$$

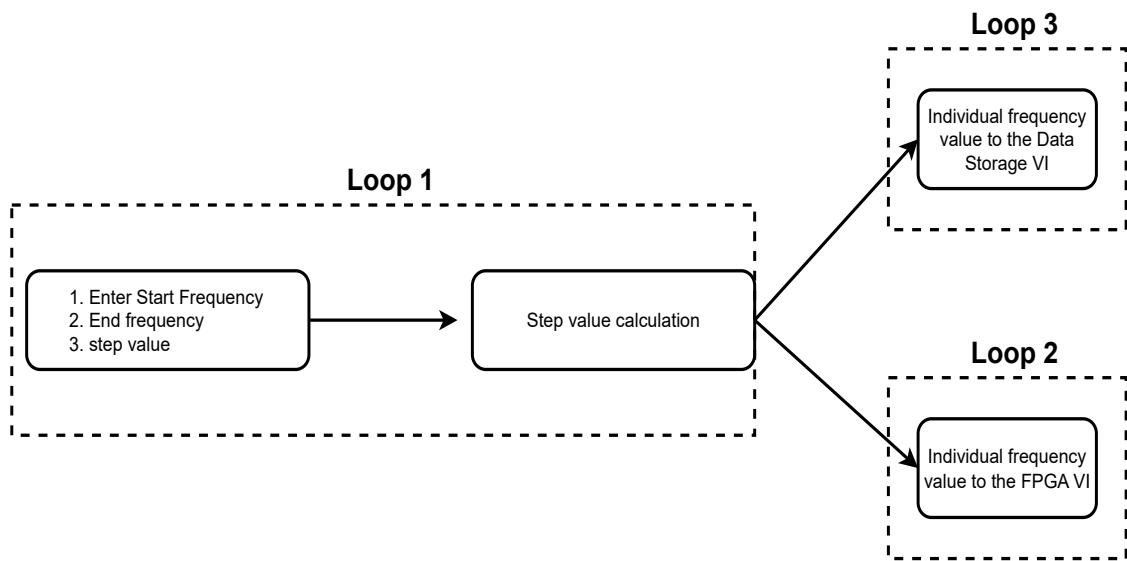


Figure 3.9: Block diagram showing the host VI Loop 1 data flow: The Loop takes the start, End frequency and the step value entered by the user and generates frequency values through the range specified for each loop iteration. The frequency values are sent to Loop 2 which is later sent to the FPGA for tuning fork actuation and also to Loop 3 to be stored in the Excel file

2. **Loop 2:** Loop 2 performs the function of data acquisition from the FIFO. This loop also configures the buffer size the FIFO requires to transfer data from the FPGA VI without loss. The information flow of loop two is shown in Figure 3.10. This is the only loop of the host VI that interacts directly with the FPGA VI. It configures the FIFO, sends the frequency values to the sine wave generator and receives the RMS values from the FPGA

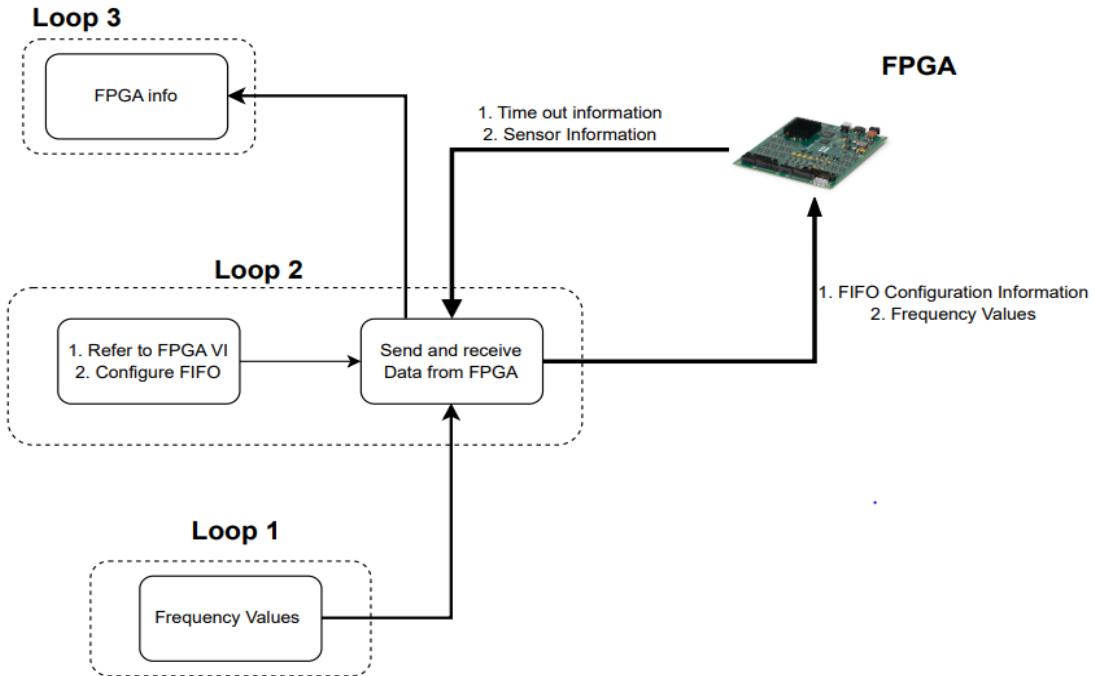


Figure 3.10: Host VI Loop 2 Data Flow: This loop controls the parameters of the FPGA VI. It receives and sends data simultaneously to and from the FPGA. The frequency values from Loop 1 are transferred to the FPGA via this loop and also the RMS Value of the oscillation Amplitude from the FPGA is sent to Loop 3. In case of any data overflow, "time out" of the is also indicated in this Loop

3. **Loop 3:** Storing the received data is the main functionality of loop 3. It acquires the frequency values from loop 1 and the waveform values from loop 2 and stores them in an excel file to plot the data and identify the resonant frequency.

All 3 loops are placed parallelly inside the bigger while loop to send, receive and transfer the data between the FPGA and the Host. The data is displayed in the Host VI front panel and stored in the user-specified file location.

Chapter 4

Frequency Sweep Results

4.1 Presettings

FPGA VI : The following steps are followed once the design is complete:

1. The RMS parameters, the sinewave generator parameters and the FIFO depth are specified according to the requirements.
2. The FPGA VI is compiled with specifications from the main menu.

Host VI :

1. In loop 2 of the Host VI (Figure 3.8), the FPGA VI is called. This must be configured before running the VI. With this step, the number of elements to read must be specified at FIFO read function. The front panel of the host VI is shown in Figure 4.1
2. Next, in loop 3, file type and storage path are specified.
3. In the front panel of the VI, the start frequency, end frequency and step values are specified. The data store button is turned on.
4. The VI is run until all the frequency steps are complete.

Plotting the obtained data : The data is stored in an Excel sheet. The first column specifies the time and date of the analog inputs as they are recorded in the Excel file, the second column contains the RMS voltage and the third column contains the corresponding frequency. The allocation is set according to the user's requirements by changing the loop 3 parameters.

Currently, the loop rates are not in sync. 2000 RMS voltage values are generated between two frequencies. This indicates that loop 2 of the FPGA VI is running 2000 times for every

4 Frequency Sweep Results

single iteration of the frequency generation loop 1. This is an indication of different loop timings for different VIs. Further controls are required in the future to bring down the loop timing of the FPGA VI such that the program does not crash and a single value of the RMS is obtained for each frequency. Now, all obtained values of the RMS waveform for each single frequency are averaged to plot the graph. From the data recorded in the excel file, it can be seen that the RMS amplitude values for each frequency remain the same or have a deviation of 0.00005 indicating that each frequency has a different oscillation amplitude of the QTF.

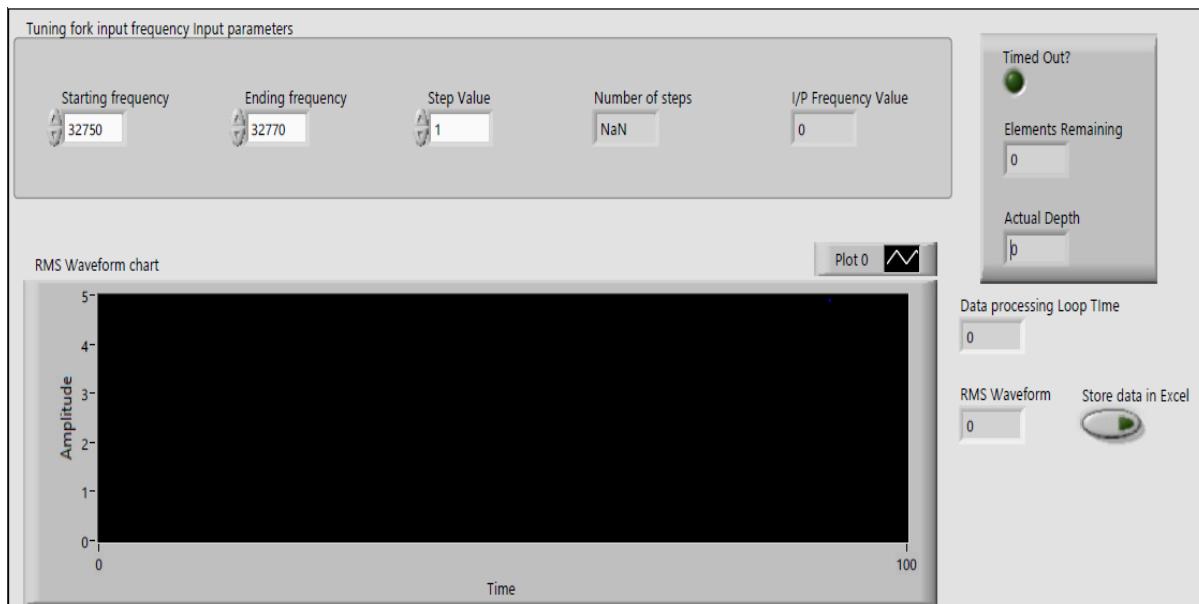


Figure 4.1: Host VI Front Panel: The starting frequency, ending frequency and step value must be entered by the user before running the VI. The timeout, elements remaining and actual depth are parameters of the FIFO, used to control the data flow. If the timeout light is on or the number of elements remaining equals the actual depth, this indicates a data overflow and hence a data loss from the FIFO. Loop time provides the time taken to complete one iteration of the data processing loop. The ‘Store Data’ button is a switch used to control data storage in the Excel file. The RMS value of the signal from the tuning fork is also displayed.

The experiment is conducted for 3 different variations of the tuning fork. a

1. Tuning fork in vacuum (with the cap on)
2. Tuning fork in atmosphere
3. Tuning fork in the atmosphere with a tip attached.

In the Excel file where the data is stored, the frequency value is stored in the second column and the RMS voltage value V_{RMS} is stored in the third column and for each frequency value, a number N of RMS voltage values are generated. Hence, an average value V_{avg} of RMS

4 Frequency Sweep Results

voltage is calculated for each frequency. This step is repeated for all the values of frequencies. The goal is to end up with one RMS voltage value for one frequency and use this to plot the graph of average RMS voltage value versus frequency.

$$V_{\text{avg}} = \frac{(\sum V_{\text{rms}})}{N}$$

The deviation of the V_{RMS} is also calculated using standard deviation. If V_1 is the first RMS voltage of a particular frequency and V_N is the last RMS voltage for the same frequency and N is the total number of RMS voltage values for a particular frequency, the standard deviation σ is calculated as follows.

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (V_i - \bar{V})^2}$$

where \bar{V} is the mean of the RMS voltage values, calculated as:

$$\bar{V} = \frac{1}{N} \sum_{i=1}^N V_i$$

MATLAB software is used to plot the processed data from the VI. The following section shows how the amplitude varies with changes in frequency for 3 different settings of QTF.

4.2 QTF in vacuum

From Figure 4.2, it can be seen that the QTF oscillates with the highest amplitude when the frequency of the analog input signal is equal to the resonant frequency value of the tuning fork. According to the experimental data, the resonant frequency value is 32762 Hz for this setup of the QTF. There is an acute increase in oscillation amplitude at 20000 Hz. This may be due to measurement error. It is also possible that the inaccuracy of the actuation signal could cause this error.

4 Frequency Sweep Results

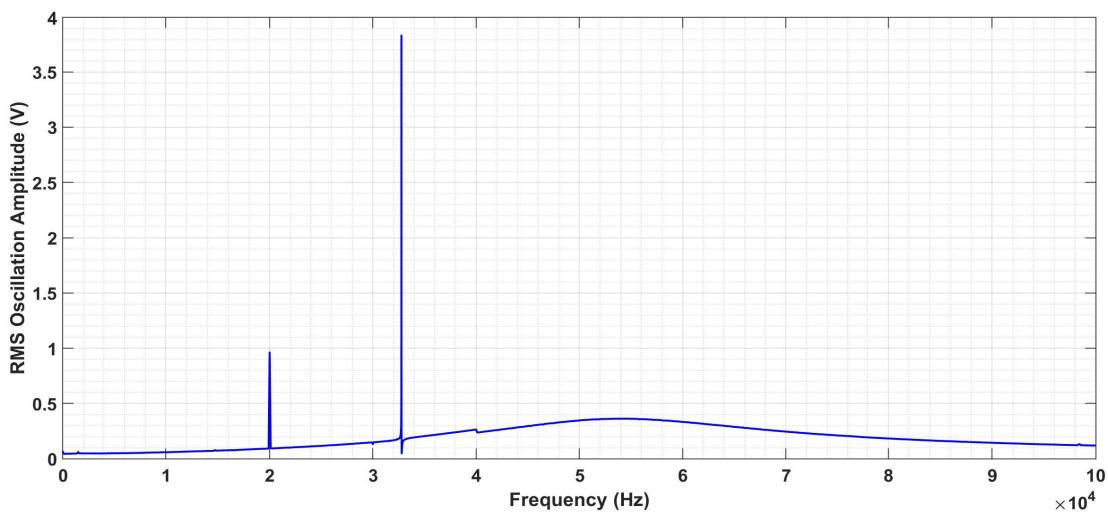


Figure 4.2: QTF in Vacuum/capped: The RMS amplitude reaches a maximum at a frequency of 32762 Hz and reaches a peak of 3.86 Volts. This indicates the first Eigenmode of the QTF

4.3 QTF in Atmosphere

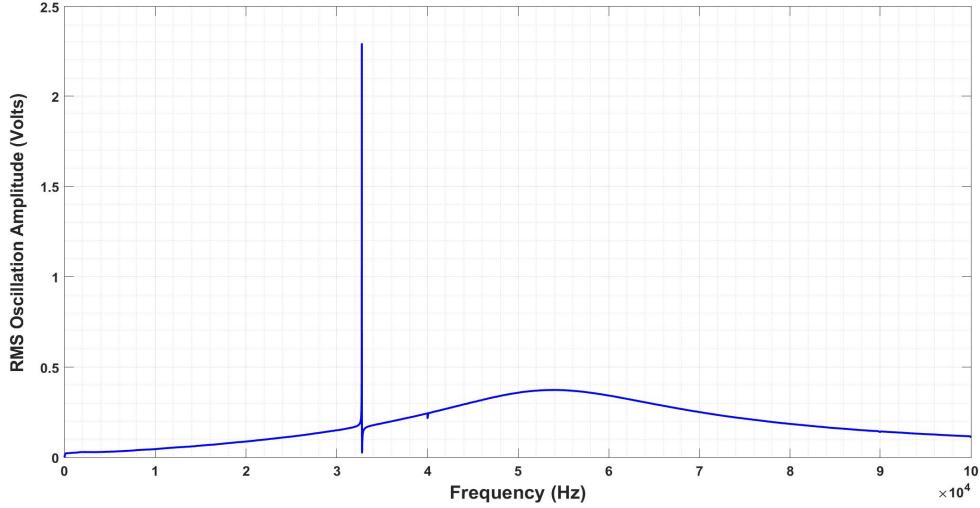


Figure 4.3: QTF in Atmosphere: The RMS amplitude reaches a maximum of 2.3 Volts at the frequency of 32758 Hz. This indicated the first Eigen mode of this variation of the QTF

When the cap of the QTF is removed and oscillation takes place in the atmosphere. It can be observed that the amplitude of oscillation decreases significantly (Figure 4.3). This can be associated with the damping that occurs due to the resistance offered by the air particles. Therefore, the quality factor of the QTF reduces and affects the sensitivity [3]. It is observed that the resonance of the QTF occurs at a frequency of 32758 Hz, which is lower than the resonance frequency in a vacuum. The shift in the frequency can also be due to the inaccuracy of signal integrity discussed in chapter 5.

4.4 QTF with wire tip attached

For the next configuration, a tungsten tip is attached to the end of one of the prongs of the QTF. Silver epoxy is used to attach the tip to the prong. The length L of the tip is 1.4 mm and the diameter D of the tip is 0.125 mm and the density ρ of the material is 19300 kg/m^3 . The Volume V and the tip's mass m are calculated using these parameters. For simplicity, it is assumed that the tip attached is a cylinder and the mass of the silver epoxy is ignored.

$$D = 0.125 \text{ mm} = 0.125 \cdot 10^{-3} \text{ m}$$

$$R = \frac{d}{2} = 0.0625 \cdot 10^{-3} \text{ m}$$

$$L = 1.4 \text{ mm} = 1.4 \cdot 10^{-3} \text{ m}$$

$$V = \pi r^2 L$$

$$= \pi \cdot (0.0625 \cdot 10^{-3} \text{ m})^2 \cdot 1.4 \cdot 10^{-3} \text{ m}$$

$$= \pi \cdot (3.90625 \cdot 10^{-9} \text{ m}^2) \cdot 1.4 \cdot 10^{-3} \text{ m}$$

$$= \pi \cdot 5.46875 \cdot 10^{-12} \text{ m}^3$$

$$\approx 1.71887 \cdot 10^{-11} \text{ m}^3$$

$$\rho = 19300 \text{ kg/m}^3$$

$$m = \rho \cdot V$$

$$= 19300 \cdot 1.71887 \cdot 10^{-11} \text{ m}^3$$

$$\approx 3.31336 \cdot 10^{-7} \text{ kg}$$

$$= 0.331336 \text{ mg}$$

Hence, a total mass of 0.331336 milligrams is added to the QTF.

4 Frequency Sweep Results

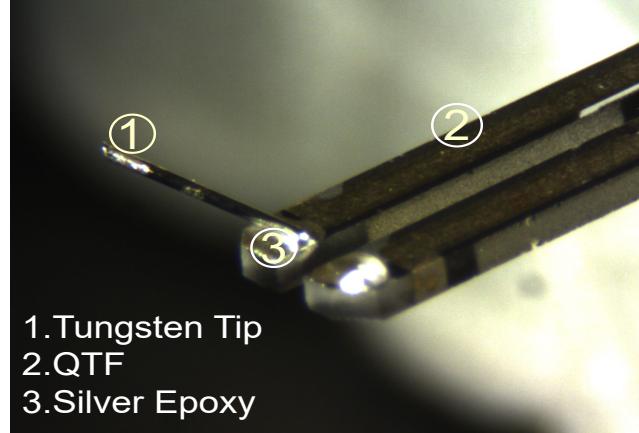


Figure 4.4: Tungsten tip is attached to one of the prongs of the QTF using silver epoxy

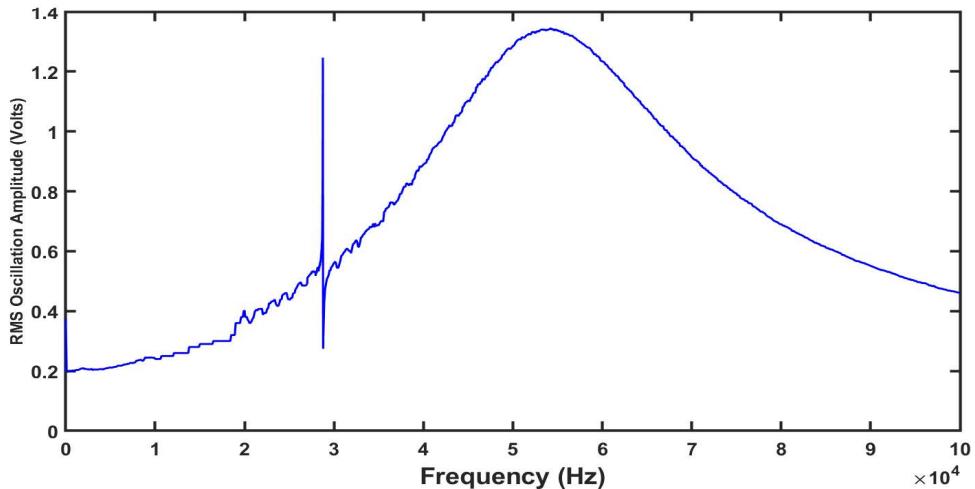


Figure 4.5: QTF in atmosphere with tip attached: The maximum RMS Oscillation values of the tuning fork in this variation at its Eigen mode is 1.26 but due to the coupling of the fork as a result of the additional mass, there is a gradual increase in the frequency till 55000 Hz

QTF with tips made of other materials such as glass find their applications in AFM for surface profile measurements [16]. Due to the mass of the attached tip, the total mass of the system increases, therefore the resonant frequency decreases. This is due to the increase in inertia of the system. An image of the QTF with a tip attached is shown in Figure 4.4.

From Figure 4.5, it can be seen that there is a sharp increase in amplitude at 28750

4 Frequency Sweep Results

Hz. This corresponds to the first eigenmode of the tuning fork. Compared to the other two variations, the peak amplitude of oscillation is low. Upon closer inspection, it is evident that the signal is not smooth and comparatively has more noise. There is also a gradual increase in amplitude after the resonance peak. This is due to the coupling of the input analog signal to the output signal of the tuning fork. The electronics on which the tuning forks are mounted cause this response. This gradual increase is seen in all three variations but it is more pronounced in the QTF with the tip attached.

Chapter 5

Conclusion

5.1 Discussion of results

From the data recorded from host VI, variations in the range of 0.01 to 0.001 in the recorded RMS voltages can be seen even though the frequency value remains the same. It can be speculated that variations in the input signal are responsible for this.

Errors and Signal integrity: The image of the input signal on the oscilloscope at frequencies (1) 10000, (2) 50000, (3) 90000 are shown in the Figure 5.1. The frequency of the input signal is not constant and there is a lack of signal integrity. It has a variation of ± 50 Hz. As frequency is increased, this variation also increases, reducing the system's accuracy. This inaccuracy of the actuation signal can be attributed to the FPGA board top-level clock used by the FPGA VI's sine wave generator and the capability of the FPGA board itself. Many different configurations of the sine wave generator were used by varying the lookup table size and amplitude resolution but none of the combinations reduced this inaccuracy. Also, from the shape of the signal generated as seen on the oscilloscope, a pure sine wave is not generated by the FPGA Board (Figure 5.1). The frequency sweep could be done only up to 100 KHz and hence, only one In-plane symmetric mode of oscillation mode could be scanned, the next mode lies around 190000Hz [3] and it is not possible to generate an accurate signal in this range using the current hardware to detect the QTF output voltage. While generating the actuation signal of the QTF, initial increment steps of 50 Hz are used and the corresponding RMS values of the oscillation are noted. Once a peak or a sudden increase in this amplitude is seen on the waveform graph of the host VI, this step size value is reduced to analyze the frequency range in detail. A thorough analysis of the frequency is possible but the maximum speed of the frequency generation loop is set at 1 second and the Excel file size is too large to repeat this for 100000 values. Hence

5 Conclusion

this method was used to measure the analog output of the QTF. From the given board, it is impossible to do an in-depth analysis by increasing the frequency steps by 0.1 Hz as the minimum step size is 1 Hz for the data type input to the frequency generator of the FPGA sine wave generator.

To summarise, there are currently 3 issues with this circuit design:

- Pure sine wave is not generated.
- Fluctuation in the frequency value of the analog signal generated by the FPGA.
- The resolution of increment step size of the frequency of the signal generated by the FPGA board used to actuate the QTF

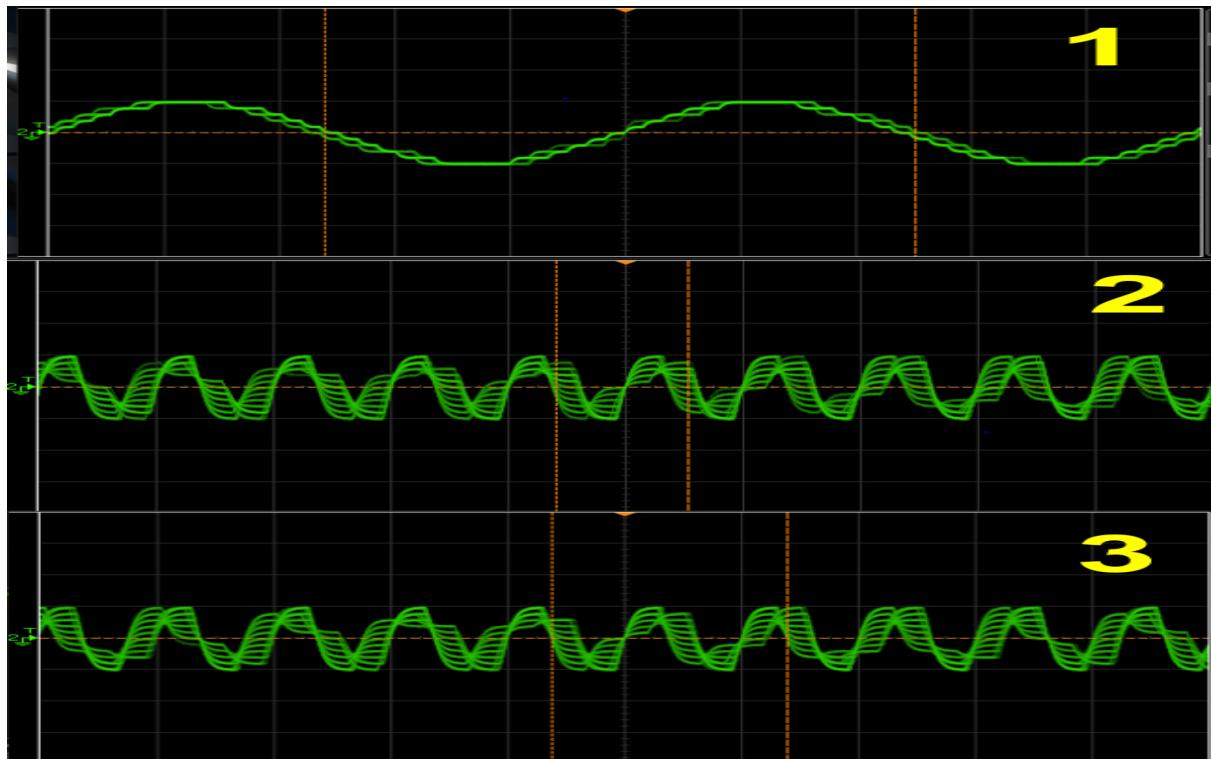


Figure 5.1: Actuation signal images as seen on the oscilloscope: With the increase in frequency, the signal integrity decreased. The frequency value keeps changing even though there is no instruction from the FPGA board to do so. Signals 2 and 3 show distorted signals at 50,000 Hz and 90,000 Hz. This condition worsens when the frequency value is above 100 kHz.

To review, an electrical setup was developed to determine the resonant frequency of the QTF. LabVIEW software is used to design a circuit and do the frequency sweep for three different configurations of the tuning fork. From the data obtained in the Vacuum capped configuration, the tuning fork oscillated with a much higher amplitude compared to the tuning

5 Conclusion

fork configuration in the atmosphere as seen from the graphs Figure 4.2 and Figure 4.3. The resonant frequency of the tuning fork in the atmosphere was also lower compared to the vacuum. This can be due to the change in quality factor due to the damping effects of air [17]. When a tungsten tip was attached to the tuning fork, the resonant frequency was decreased by approximately 4000 Hz and reached 28740 Hz and the peak RMS Voltage value also decreased to 1.23 Volts. The increase in amplitude after the resonance peak is due to the coupling of the input and output signals of the tuning fork. This effect is most enhanced in the QTF with a tip attached.

5.2 Scope for improvement

From the signal generated as seen on the Oscilloscope Figure 5.1, it can be seen that further optimization of the FPGA VI is required to improve the accuracy and reduce distortions of the actuation signal generated by the sine wave generator of the FPGA. However, further experimentation and trials must be done to determine the exact reason for the distorted signals generated by the FPGA board. From the data recorded in the Excel file, it is shown that the FPGA signal recording loop and the signal processing loop of the host VI are not perfectly in sync. This is due to the different loop rates of the two loops. Loop rate refers to the time taken to complete one iteration of the loop. If one of the loops runs faster, then it processes and transfers data faster than the neighboring loop. Currently, this does not cause any issues while recording the data but there is a possibility of error when the frequency sweep range and the loop rates are increased. A large portion of this problem can be improved by using more powerful hardware capable of running the loops faster and parallel to each other but some steps can be implemented by the user to improve this desynchronization issue. One of the methods is discussed here.

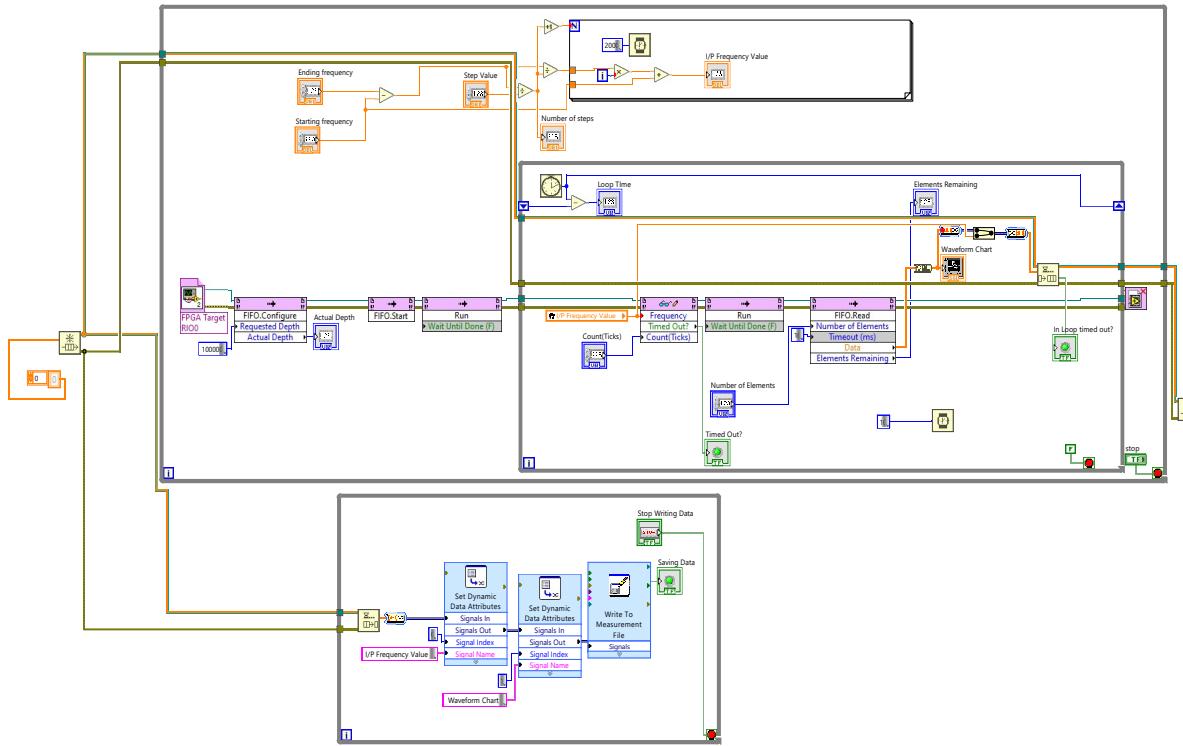


Figure 5.2: Producer Consumer Loop design architecture is used to transfer data between loop 2 and loop 3 of host VI. The usage of Local variables is avoided to store the data in the Excel file in this design. Hence, there is a buffer available to avoid data loss due to desynchronization between the different loops. Consequently, it would be possible to store data even with Loops running at different rates

The local variables used for the inter-VI communication of the host VI do not behave as a buffer and hence there is a chance of losing data while the transfer takes place. One solution for this can be the use of an architecture called "Producer-Consumer" architecture in LabVIEW. The circuit shown in the Figure 5.2 is an iteration of the host VI when used with a producer-consumer architecture. The data storage is the current issue for this circuit and it can be further improved using timers and error controls. Despite this, a software and electrical set has been developed to automatically and swiftly determine the resonance frequencies of different variations of the QTFs, even though there is still room for improvement.

5.3 Outlook

Even though the current iteration of the hardware circuit has many errors, by changing a few components and using additional hardware, the signal integrity can be improved by a large

5 Conclusion

margin. For instance, by integrating a function generator capable of communication with the FPGA VI, host PC can be used and the FPGA board to control the frequency sweep that generates a pure sine wave. This can reduce the inconsistency and the frequency fluctuation seen in the current circuit and this would also solve the issue of generating a pure sine wave. Therefore, any synchronization issue of the loops would be eliminated using this iteration of the circuit. By the addition of a laser interferometer to the circuit, it would be possible to record the mechanical oscillation amplitude and hence, measure the deflection of the beam through a particular frequency range or at the resonant frequency. To conclude, based on the requirements, the hardware can be configured and integrated with the current design.

Bibliography

- [1] N. Instruments, *Data Transfer Using FIFOs*, 2024, accessed: 2024-05-23. [Online]. Available: <https://www.ni.com/docs/de-DE/bundle/labview-nxg-fpga-targets/page/data-transfer-using-fifos.html>
- [2] Y. Seo and W. Jhe, “Atomic force microscopy and spectroscopy,” *Reports on Progress in Physics*, vol. 71, no. 1, p. 016101, 2007.
- [3] J.-M. Friedt and É. Carry, “Introduction to the quartz tuning fork,” *American Journal of Physics*, vol. 75, no. 5, pp. 415–422, 2007.
- [4] M. Lee, B. Kim, S. An, and W. Jhe, “Dynamic responses of electrically driven quartz tuning fork and qplus sensor: A comprehensive electromechanical model for quartz tuning fork,” *Sensors*, vol. 19, no. 12, 2019. [Online]. Available: <https://www.mdpi.com/1424-8220/19/12/2686>
- [5] C. R. Blanchard, “Atomic force microscopy,” *The chemical educator*, vol. 1, pp. 1–8, 1996.
- [6] S. Kasas, N. Thomson, B. Smith, P. Hansma, J. Miklossy, and H. Hansma, “Biological applications of the afm: from single molecules to organs,” *International journal of imaging systems and technology*, vol. 8, no. 2, pp. 151–161, 1997.
- [7] S. K. Koc, “Applications of atomic force microscopy in textiles,” *Journal of Engineered Fibers and Fabrics*, vol. 10, no. 1, p. 155892501501000118, 2015.
- [8] S. Liu and Y. Wang, “A review of the application of atomic force microscopy (afm) in food science and technology,” *Advances in food and nutrition research*, vol. 62, pp. 201–240, 2011.
- [9] J. Kim, D. Won, B. Sung, S. An, and W. Jhe, “Effective stiffness of qplus sensor and quartz tuning fork,” *Ultramicroscopy*, vol. 141, pp. 56–62, 2014.

Bibliography

- [10] R. Lin, J. Qian, Y. Li, P. Cheng, C. Wang, L. Li, X. Gao, and W. Sun, “Equivalent electromechanical model for quartz tuning fork used in atomic force microscopy,” *Sensors*, vol. 23, no. 8, p. 3923, 2023.
- [11] H. Edwards, L. Taylor, W. Duncan, and A. J. Melmed, “Fast, high-resolution atomic force microscopy using a quartz tuning fork as actuator and sensor,” *Journal of applied physics*, vol. 82, no. 3, pp. 980–984, 1997.
- [12] University of Cambridge, “Dissemination of it for the promotion of materials science (doitpoms),” Online, 2024, available: <https://www.doitpoms.ac.uk>.
- [13] O. E. Dagdeviren and U. D. Schwarz, “Numerical performance analysis of quartz tuning fork-based force sensors,” *Measurement Science and Technology*, vol. 28, no. 1, p. 015102, 2016.
- [14] R. Garcia and R. Perez, “Dynamic atomic force microscopy methods,” *Surface science reports*, vol. 47, no. 6-8, pp. 197–301, 2002.
- [15] National Instruments, “Ni support,” <https://www.ni.com/de/support.html>, accessed: 2024-05-31.
- [16] Y.-L. Chen, Y. Xu, Y. Shimizu, H. Matsukuma, and W. Gao, “High quality-factor quartz tuning fork glass probe used in tapping mode atomic force microscopy for surface profile measurement,” *Measurement Science and Technology*, vol. 29, no. 6, p. 065014, 2018.
- [17] M. Olivieri, A. Zifarelli, G. Menduni, M. Di Gioia, C. Marzocca, V. M. Passaro, A. Sampaolo, M. Giglio, V. Spagnolo, and P. Patimisco, “Influence of air pressure on the resonance properties of a t-shaped quartz tuning fork coupled with resonator tubes,” *Applied Sciences*, vol. 11, no. 17, p. 7974, 2021.