Localized Harmonic Motion Imaging

by

Ping Gong

Supervised by: Dr. Laura Curiel, and Dr. Samuel Pichardo

A thesis submitted in partial fulfillment of the requirements of the
M.Sc.Eng degree in
Electrical and Computing Engineering
Faculty of Engineering
Lakehead University
Thunder Bay, Ontario

Thunder Bay, Ontario, Canada
April 26th, 2012

Abstract

Localized Harmonic Motion (LHM) Imaging is a new technique of ultrasound imaging which uses the localized stimulus of the oscillatory ultrasonic radiation force as produced by a modulated signal, and estimates the resulting harmonic displacement in the tissue in order to assess its underlying mechanical properties. This method can be highly localized and is considered as a non-invasive modality.

In this thesis we first present the background information for LHM imaging and we compare this technique to other tissue mechanical properties imaging techniques.

We then describe a setup for LHM induction and how the data is acquired and processed. We first focus on the transducer configuration, its characteristics and the housing built to combine the transducers, then the alignment of these transducers so they are confocal and can induce and detect motion in tissues, and finally we describe the local harmonic motion experiment setup including a supporting system and the induction/detection module.

One of the most critical stages is the acquisition of the signal, since signals acquired by the imaging transducer always contain different sources of noise such as acoustic (standing waves, reflection from the tank, mechanical cross-talk between the transducers) and electric noise (electric cross-talk, noise of the high power amplifier) that we need to filter. Electronic filters were designed and implemented into our LHM experiment system. Additionally, digital filters were designed to further improve the performance of the system. We applied several kinds of digital notch filters (finite impulse response (FIR) and infinite impulse response (IIR) classes) and conduct analysis on the performance when obtaining LHM displacement information.

After finishing the filtering and the setup, we performed LHM displacement experiments. We analyzed the obtained displacements as well as the noise observed in the final displacement waveforms, and the influence of analog and digital filters on the displacement detection. We finally measured the displacements induced by LHM on samples with different Young modulus and were able to differentiate them by the amplitude of the motion.

Finally, we performed optimizations on the algorithm for LHM displacement calculations. Due to the large amount (462) of RF signals, it will typically take around 1h for a 41x41 points image. It was found that the digital filter was the most time consuming part of the processing and it was parallelized using graphics processing unit (GPU).

Acknowledgment

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Meaning
ARFI	Acoustic Radiation Force Imaging
CPU	Centre Processing Unit
FUS	Focused Ultrasound
FIR	Finite Impulse Response
GPU	Graphics Processing Unit
IIR	Infinite Impulse Response
LHM	Localized Harmonic Motion
MRI	Magnetic Resonance Imaging
RF	Radio Frequency
SSI	Supersonic Shear Imaging technique
US	Ultrasound
USVA	Vibro-Acoustography

Introduction

Localized Harmonic Motion (LHM) Imaging is a new technique of ultrasound imaging which uses the localized stimulus of the oscillatory ultrasonic radiation force as produced by a modulated signal, and estimates the resulting harmonic displacement in the tissue in order to assess its underlying mechanical properties. This method can be highly localized and is considered as a non-invasive modality compared to x-rays or other types of ionizing radiation.

We propose to detect the LHM displacements induced by modulated signal and compare the amplitude on samples with different stiffness. This thesis covers background information such as the techniques that have been developed in an effort to estimate mechanical properties of tissues and the advantage of LHM over other techniques. Also provides a description of the LHM experiment setup which includes transducer configuration, transducer alignment and the complete setup.

We also present a series of filters designed to control the noise that is intrinsic to the acquisition of the signals used for LHM imaging. This section is divided into two separate parts: analog and digital filters. We then present the results for the displacements obtained after LHM induction inside samples and an analysis of the displacement amplitude as a function of the sample stiffness. The thesis also discusses the optimization of the computational algorithm used for displacement calculations.

Finally, we briefly present the future work for the project, which includes the next steps as well as suggestions for running the experiment in the future.

1. Background

1. 1 Technologies

1.1.1 Introduction

On standard ultrasound echo imaging, due to the isoechoic nature of lesions and their poor contrast with neighbouring tissue, a lesion may remain undetected. Thus, there has been an intense effort to improve the sensitivity and specificity of cancer detection using ultrasound modalities. An area of considerable interest is the use of tissue elasticity [1]. Imaging of the elastic properties of tissue provides new information which is collateral to tissue pathology [2].

For centuries, physicians have used palpation as an important diagnostic tool. The efficacy of palpation is based on the fact that many diseases cause changes in the tissue's mechanical properties [3]. While palpation is understandably simple, it is nevertheless just a qualitative assessment and can only be applied to superficial organs. Therefore we know that diseases result in an increase in stiffness or elastic modulus (like Young modulus) of the tissue which allows us to differentiate malignant tumours from possibly benign conditions. Several techniques have been developed in an effort to estimate mechanical and acoustic properties of tissues [4]. We will introduce here four of these techniques: elastography (dynamic and transient); vibro-acoustography (USVA); acoustic radiation force imaging (ARFI) and local harmonic motion (LHM).

1.1.2 Basic concept

If one or more of the tissue elements has a different stiffness, the level of strain in that element will generally be higher or lower; a harder tissue element will generally experience less strain than a softer one [3]. To measure the stiffness parameters of tissue a mechanical stimulus must be propagated into the tissue, and means for detecting the resulting internal motions must be provided. Ultrasound imaging can provide these means.

Tissue elasticity imaging methods based on ultrasound can be divided depending on the mechanical stimulus:
1) quasi-static compression from which strain can be estimated; or
2) quasi-continuous vibration from which tissue response can be analysed [5].

In most of the ultrasound methods, we send an acoustic wave to the tissue and use this wave to measure object motion using principles similar to radar and sonar. Ultrasound

presents advantages for measuring tissue characteristics since it is a non-invasive technique with high spatial resolution [6].

1.1.3 Elastography

Elastography is an emerging medical imaging method that evaluates the tissue stiffness, which is dependent on the molecular component and the corresponding microstructure of biological tissue and correlates with pathology [7].

Elastography consists of applying a pressure on the examined medium and in estimating the induced strain distribution by tracking the tissue motion [3]. When an elastic medium, such as tissue, is compressed by a constant uniaxial stress, all points in the medium experience a resulting level of longitudinal strain whose principal components are along the axis of compression. A harder tissue element will generally experience less strain than a softer one. The longitudinal axial strain is estimated in one dimension from the analysis of ultrasonic signals obtained from ultrasound signals. This is achieved by acquiring a set of digitized echo lines from the tissue region of interest while compressing the tissue by a small amount (about 1 percent or less of the tissue depth). Congruent echo lines are then subdivided into small temporal windows which are compared pairwise by using cross-correlation techniques from which the change in arrival time of the echoes before and after compression can be estimated. Due to the small magnitude of the applied compression, there are only small distortions of the echo lines [5] (Figure 1).
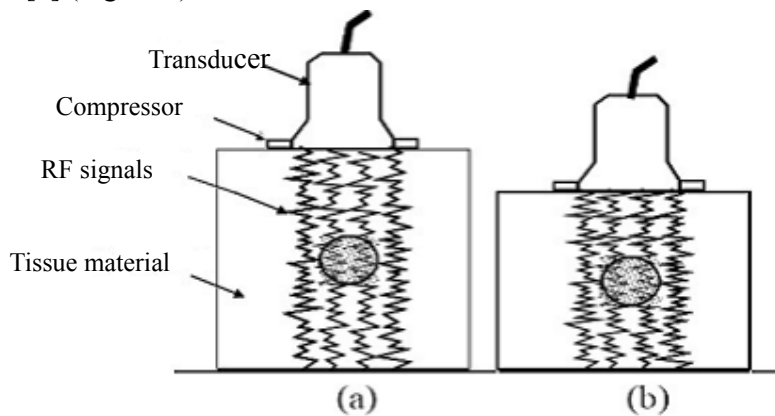
Figure 1. The principle of elastography: tissue is insonified a) before and b) after a small uniform compression [8].

A stiffer tissue element will generally experience less strain than a softer one [3]. Therefore we can differentiate harder parts and softer ones depending on how much displacement we can induce under the same pressure. The abnormal tissues have an increase in stiffness or elastic modulus which is the main idea of elastography.

Elastography is based on the following assumptions:

1. The load applied to the tissue can be considered to be static.

2. The data acquisition time is very short compared to the time over which loading changes.

3. The strain in the tissue is very small (on the order of 1% or less) and can be considered to be linear [9].

The tissue strain is typically estimated from the gradient of tissue displacements. The local tissue displacements are estimated from the time delays of pre- and post-compression echo signals using cross-correlation analysis. More details on cross-correlation techniques will be discussed in Chapter 5.

There are two variations to elastographic techniques: transient and dynamic elastography.

1.1.3.1 Transient elastography:

The goal of transient elastography is to create high-resolution shear stiffness images of tissue. Shear stiffness is targeted because the shear wave speed is 2 to 4 times larger in abnormal tissue than in normal tissue [10].

In transient elastography, tissue initially at rest is excited with a broadband single pulse at the boundary or along a line in the interior. Typically, the pulse has a central frequency of 50–200 Hz. This pulse creates two different types of waves that propagate into the medium: compression and shear waves. The compression wave displaces tissue in the same direction that the wave propagates, it is a fast wave that travels at approximately the speed of sound in tissues, 1500 ms^{-1}.The shear wave displaces tissue orthogonally to the direction that the wave propagates. This wave is a very slow wave that travels at 3 ms^{-1} [10].

Transient elastography uses mechanical stimulation to generate shear waves and the echoes are collected by an ultrasound transducer. The resulting shear displacements are estimated using ultrasonic correlation based algorithms. Strain image can be obtained eventually through applying a classical differentiation algorithm to the estimated displacements. Shear velocity is the most important parameter, since it is directly related to elasticity and stiffness [7]. The final goal then is to use the acquired signals to reconstruct the shear wave speed [10].

1.1.3.2 Dynamic Elastography

It is also known as supersonic shear imaging (SSI). SSI is based on the combination of radiation force induced by an ultrasonic beam and an ultrafast imaging sequence capable of catching in real time the propagation of the resulting shear waves. Then by measuring the shear wave, the complete 2D elasticity map can be retrieved [11].

The first step is the generation of a shear wave at the surface of the media by means of the radiation force (as opposed to a mechanical external excitation as explained above) [11]. Immediately after the generation of the shear wave, the system switches into an ultrafast imaging mode to acquire a movie of the propagating shear wave. Then by using time of flight algorithm the shear velocity is retrieved to form the image. In addition, the amplitude of the shear wave can allow us to determine its attenuation. The computation of a complete bi-dimensional (2D) shear elasticity (μ) map can be obtained by using:

$$\mu = \rho \cdot V_s^2 \tag{1.1.1}$$

Where ρ represents the density of the media and V_s is the shear wave velocity [13]. Also, a standard Doppler ultrasound technique is used to measure the particle motion (and therefore the displacement) induced. The received frequency shift (Δf) caused by the Doppler effect is obtained with:

$$\Delta f = \frac{2f}{c} v_0 \cos\theta \tag{1.1.2}$$

Where θ is the angle between propagation axis and the shear wave, v_0 is the particle speed. Therefore the information of displacement of the tissue can be obtained which will also allow detecting the tumor.

However, there are some limitations of this method like peaks and lows in the wave also depend on the geometry of tissue which is difficult to determine. Moreover, high frequency shear waves are not suitable for use as a probe, because the attenuation is high.

1.1.4 Vibro-Acoustography (USVA)

Using a dynamic force is a way of measuring the dynamic characteristics of a material [16]. Vibro-acoustography uses ultrasound radiation force to remotely exert a localized oscillating stress within an object. A portion of the object vibrates sinusoidally in a pattern determined by its elastic properties. The acoustic emission field resulting from the vibration is then detected by an external hydrophone and used to form an image that represents both the ultrasonic and low-frequency (kHz range) mechanical characteristics of the object. USVA is sensitive enough to detect object motions on the order of nanometers [17] (Figure 2).
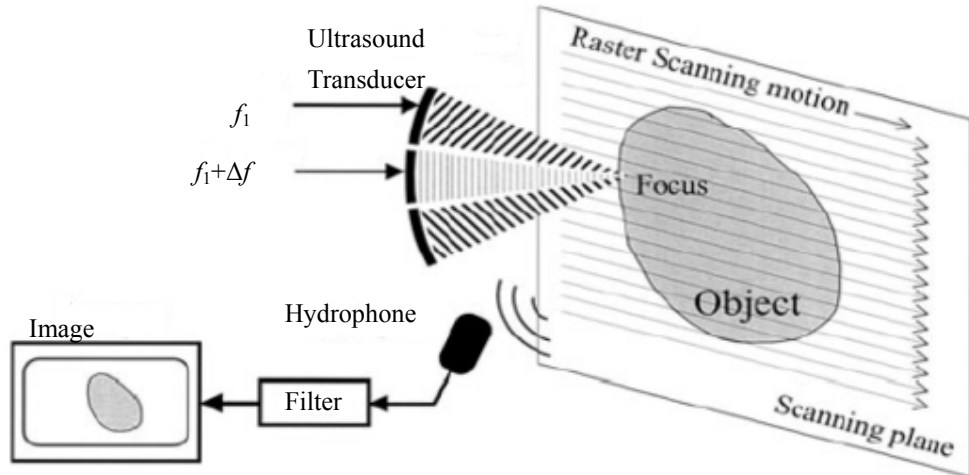
Figure 2. Vibro-Acoustography system [18].

A confocal transducer produces two continuous waves of slightly different frequencies (f_1 and $f_1+\Delta f$) that intersect at the object. The vibrations of the object are detected by the hydrophone (microphone or vibrometer). In order to form a 2D image, the whole area is covered by raster scanning of the transducers. Whereas, the frequency (f) is in order of MHz, the difference frequency (Δf) is in order of kHz [6].

The waves interfere and generate a wave that is amplitude modulated by their difference frequency (Δf). This is accomplished by projecting two coaxial confocal continuous-wave (CW) ultrasound beams on the object. An amplitude-modulated field is produced only at the interference region of the two unmodulated beams around their focal areas, resulting in a small resolution cell [17]. Object vibration results in an acoustic field that is related to object shape, size, and viscoelastic properties which we use to differentiate the abnormal part. An example of a VA image is shown in Figure 3.
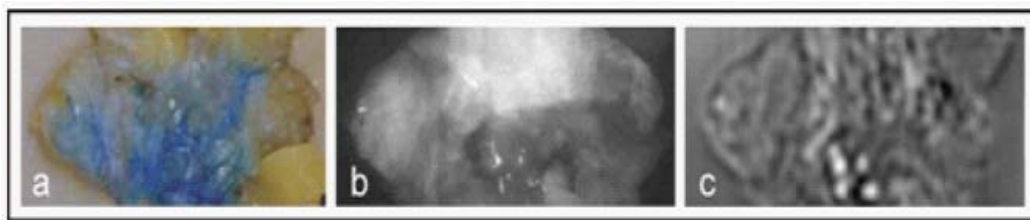
Figure 3. Vibro-acoustography of the second specimen. (a) Photograph of the tissue specimen mounted on the scanning bracket. (b) X-ray mammogram of microcalcifications at the center of the image. (c) Vibro-acoustography of (a). Microcalcifications can be seen as bright spots [19].

1.1.5 Acoustic Radiation Force Imaging (ARFI)

Acoustic Radiation Force Impulse (ARFI) imaging is another dynamic method used to generate images of the mechanical properties of tissue [20]. This method offers some advantages over those requiring external excitation, because the localized force is applied directly to the region-of-interest (ROI), thus, smaller stresses are used and the impact of external boundaries is reduced [21].

ARFI imaging uses short-duration ($<100\mu s$), high intensity, acoustic pulses to generate localized, micron-scale displacements in tissue, and these displacements are tracked using ultrasonic, correlation-based methods. Images of two-dimensional regions of interest are generated by sequentially interrogating multiple lateral locations [22].

The response of tissue to focused radiation force excitation is complex and depends upon tissue geometry, tissue mechanical and acoustic properties, in addition to the spatial distribution, magnitude and duration of the applied force. Softer tissue move farther, take longer to reach peak displacement and recover more slowly than stiffer tissues [21].

An ARFI imaging system uses two types of acoustic beams: a high-intensity pushing beam and a conventional B-mode tracking beam. In each location, a reference "track" beam is fired and the received echoes are stored; then, the high-intensity pushing beam is transmitted to displace the tissue. This is followed by several track beams to allow observation of the tissue response through time [21]. In general, tissue displacement is inversely related to tissue stiffness and tissue recovery response is related to tissue viscoelastic properties [20].

Figure 4 shows a comparison of B-mode (which a linear array of transducers are used to simultaneously scan a plane through the body that can be viewed as a two-dimensional image on screen) and ARFI images for normal proximal colon tissue. The bottom row shows an enlarged region of the upper right ARFI displacement image (left), and the normalized axial displacements averaged across the region indicated by the vertical dashed lines (right). The regions of decreased displacement in the submucosal layer (15 - 17 mm) may correspond to stiffer lymphoid aggregates. Better layer distinction with ARFI imaging may allow for more accurate staging of tumor invasion [22].
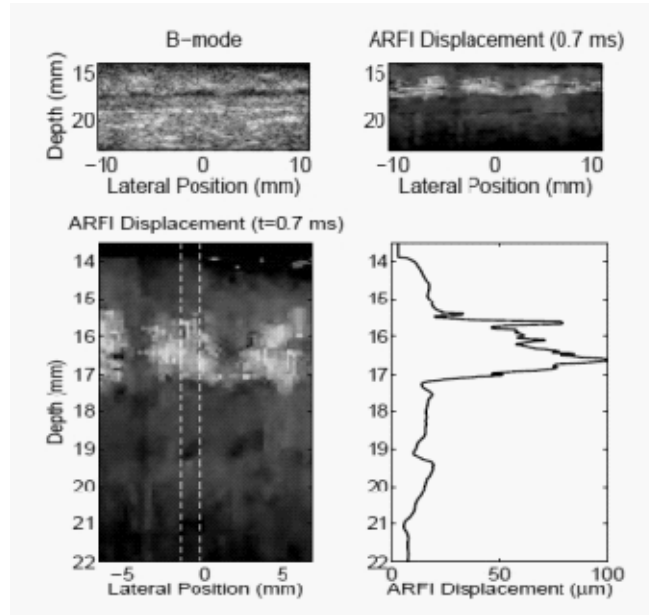
Figure 4. A comparison of B-mode and ARFI images for normal, proximal colon tissue [22].

1.1.6 Localized Harmonic motion (LHM)

Interested in obtaining images of samples with different stiffness, a new technique called localized harmonic motion (LHM) has been utilized in our research in order to differentiate tissue by its elastic properties.

In this section, we will discuss the advantages of utilizing LHM, another dynamic method that performs RF signal tracking in order to estimate the localized oscillatory motion resulting from the harmonic radiation force produced by two overlapping focused beams or a single amplitude modulated beam. Due to the highly localized and harmonic nature of the estimated response, this technique may be proven highly suitable for accurate estimation of the elastic modulus variation in tissues due to disease [4]. This method has also been used to detect and control thermal ablation [24].

Concept:

LHM imaging is a recently developed technique that evaluates the mechanical properties of soft tissue [23]. The main principle is the induction of an oscillatory, remotely applied; harmonically varying radiation force that causes tissue displacement that can be measured. The time varying radiation force can be induced by overlapping two ultrasound beams radiating at slightly different frequencies or by radiating the tissues with amplitude-modulated single-frequency ultrasound beam [23]. The ultrasound beam induces a time varying force that induces oscillation of tissue at the frequency difference between the two

beams or at the modulation frequency of the single beam.

The actual movement of the tissues is tracked by another diagnostic ultrasound beam. As the estimated response is directly obtained from the tissues undergoing vibration, the information is more dependent on the tissue characteristics at the focus than on the surrounding tissue [24].

Detection Methods:

A setup for measuring localized harmonic motion is shown in Figure 5:
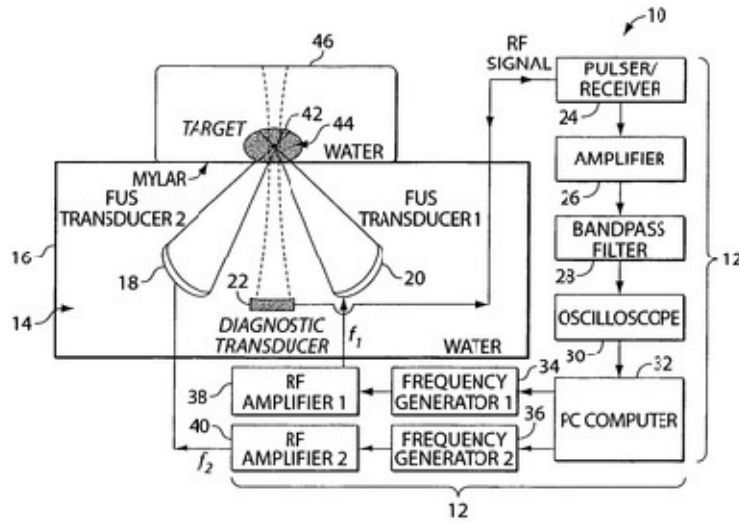
Figure 5. Localized Harmonic Motion Imaging diagram [25].

The localized harmonic motion imaging system in Figure 5 (10) includes an ultrasound generator/receiver subsystem (12), a transducer set (14), and a tank (or other coupling technique) (16). The set (14) includes two focused ultrasound (FUS) transmitting transducers (18), (20), and a diagnostic (imaging) transducer (22). The subsystem (12) includes a pulser/receiver (24), an amplifier (26), a band pass filter (28), an oscilloscope (30), a computer (32), frequency generators (34), (36) and radio frequency (RF) amplifiers (38), (40). The system (10) is configured to image a target (42) and more particularly a region (44). The tank (16) is filled with degassed water, and is separated from a target tank (46).

In this diagram, the radiation force is applied using two overlapping beams radiating at slightly different frequencies. The harmonic motion is estimated at different snapshots of the motion (t_1, t_2, etc.) using cross-correlation of RF ultrasonic signals acquired at the same location undergoing vibration by a separate ultrasound beam. This method is distinctly different from methods such as transient elastography or USVA that estimate motion during relaxation of the tissue after application of the force instead of during its application [4].

9

Cross-correlation technique is utilized to calculate the harmonic motion displacements (the details will be talked about in Chapter 5). RF line tracking at different instants (t_1, t_2, etc.) acquired at the focus of the diagnostic transducer yields precise displacement estimates and identifies the characteristics of the locally induced vibration [4].

Figure 6 shows an example of an image of the Young modulus distribution (Figure 6(a)) and the displacement image (Figure 6(b)). The inclusion is clearly depicted in the displacement image.
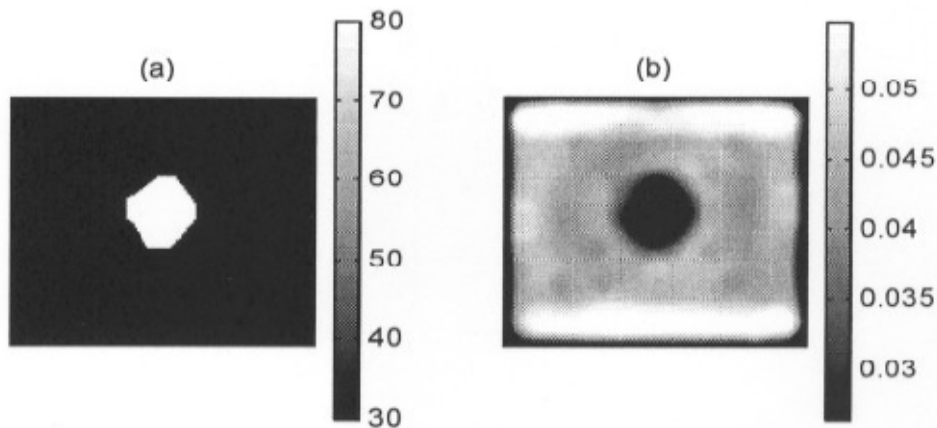
Figure 6. (a) True Young modulus (kPa) image and (b) Mean-squared estimated displacement (mm) image. The inclusion is clearly depicted [4].

Application of LHM:

(a). Imaging

As we showed in Figure 6, LHM applies a remote, harmonically variable radiation force deep inside the tissue and estimates the resulting harmonic displacement using cross-correlation techniques. The high precision of the cross-correlation techniques permits the estimation of micron-level displacements without the interference of standing waves, tank resonance and ambient noise, which may contaminate the amplitude of the signal. Most importantly, the amplitude of the displacement estimated decreases with hardness in an exponential fashion and can yield an accurate image for lesion localization, rendering this method highly sensitive for differentiation of tissues based on their hardness [4].

(b). Focused Ultrasound (FUS) therapy

As we discussed before, due to the fact that the displacements in LHM technique are measured during application of the acoustic radiation force, this method can be applied for real-time monitoring of temperature-related stiffness changes of tissues during FUS or other thermal therapies.

High levels of acoustic energy at the focus can cause temperature elevation that is sufficient to coagulate tissues (thermal lesions), while the surrounding tissues remain relatively unheated. The ability of FUS to cause irreversible damage in tissues has received attention from researchers as a potential technique for non-invasive cancer treatment [26].

It has been proved that during the FUS exposure a drop in the maximum amplitude value is observed and a threshold value was associated to the formation for a thermal lesion. A series of controlled thermal exposures was performed by stopping the exposure when the threshold value in LHM amplitude was reached [24] (Figure 7).
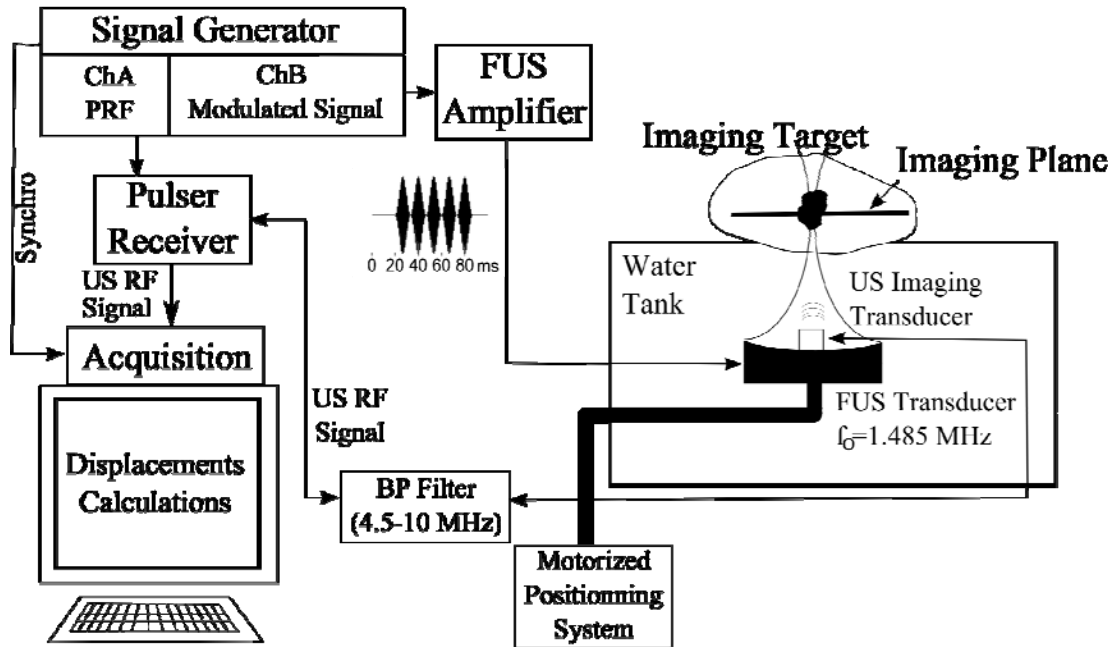
Figure 7. Experimental setup used for LHM measurements during FUS exposures. The FUS and diagnostic transducer assembly are placed in a water tank facing up, with the target tissues on top of the tank [24].

In this setup, a single-element FUS transducer operating was used to both induce the harmonic motion and create the lesions in tissue [24]. During FUS lesion formation, the harmonic motion amplitude drops when tissues harden (Figure 8) [27].
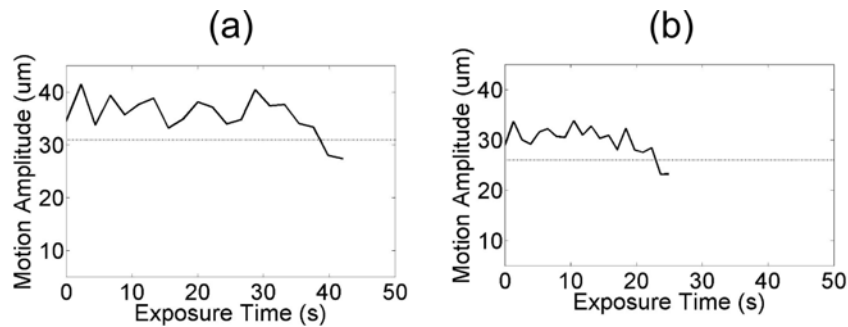
Figure 8. Amplitude of the motion during the FUS exposure for three different locations [27]. The amplitude drops during the exposure when the tissue hardens due to thermal ablation.

This system was used to control the exposure by stopping after the amplitude reached a threshold and the coagulation was confirmed later in histology [27].

1.2 Industrial/Clinical Applications:

Tissue elastic properties imaging has been integrated into commercial machines in clinical medicine such as 3D Guys Company uses Vibro-Acoustography for bone imaging and evaluation of soft tissues. While NDT Company produces ProbeScope machine due to the technique of Acoustography as shown in Figure 9.
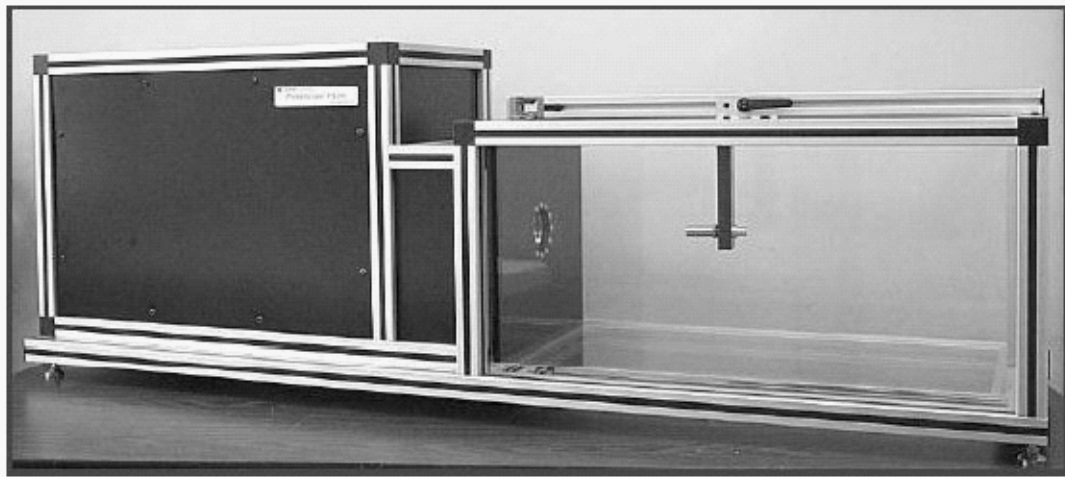
Figure 9.   The ProbeScope^{TM}, model PS200

Siemens has announced that their Acuson S2000 ultrasound system is the first medical device to implement Acoustic Radiation Force Impulse (ARFI) technology (Figure 10).

Figure 10. Acuson S2000 ultrasound system

Philips applies elastography on the iU22 machine to support diagnostic on an easy to use, robust, and highly sensitive tool. They use this method in breast imaging (Figure 11).

Figure 11. iU22 machine

1.3 Conclusion

Table 1 shows the comparison of these techniques: elastography, vibro-acoustography, radiation force, localized harmonic motion.

Techniques	Elastography	Vibro-Acoustography (USVA)	Radiation Force (ARFI)	Localized Harmonic Motion (LHM)
Characteristic	Mechanical	Acoustic	Acoustic	Acoustic
Method	Static	Dynamic	Dynamic	Dynamic
Area	Global	Local	Local	Local
Measurement	Displacement	Amplitude	Amplitude	Amplitude

Table 1. Comparison of these four techniques

Elastography differs from other methods since the stress applied to the tissue is not vibratory, but rather quasi-static. This tends to avoid problems due to reflections, standing waves and mode patterns which may be set up in the tissue and which may interfere with quality image formation [28]. However, the required external tissue compression may complicate positioning and reference during treatment planning and guidance [26]. And a drawback of static data is that, without knowledge of stress boundary conditions, the elastic modulus cannot be uniquely reconstructed [29].

The vibro-acoustography method uses two confocal ultrasound beams to generate a localized oscillatory radiation force in tissue. USVA promises applications in two general areas: medical imaging and material evaluation [19]. It produces instant high contrast images very sensitive to calcifications and detection of hard objects. However, the effect of

temperature-induced acoustic changes (e.g., speed of sound change) could not be separated from the effect of mechanical changes [26].

ARFI induces a brief localized radiation force and images the tissue response immediately after force application. ARFI images portray complementary information to conventional B-mode images. In media with structures that have greater elastic modulus contrast than acoustic contrast, ARFI images portray greater contrast [21]. A limitation of this method is that currently, tissue displacement cannot be monitored during force application due to the use of pushing and tracking beams at different times [26].

The technique we proposed here is LHM which has the advantage of using a focused beam, thus a very small focus zone can be achieved deep inside the tissue for imaging. Also, the technique has the ability of real-time synchronous monitoring of temperature-induced variation in tissue mechanical properties in a fully integrated system. By monitoring the LHM response in real time, the relative tissue stiffness changes during thermal treatment can be reliably indicated [25]. Further study of this method will include stiffness, viscoelastic properties measurements, as well as temperature and coagulation mapping, both ex-vivo and in-vivo [30].

2. Localized Harmonic Motion (LHM) Experiment Setup

2.1 Introduction:

This chapter describes a setup for LHM induction and how the data is acquired and processed. It is divided into 3 parts to ease the explanation of the entire system, including:

1. Transducer configuration: characteristics of two transducers and the housing we built to combine the FUS transducer and imaging transducer together;
2. Transducer alignment: the experiment to make the two transducers confocal;
3. Local Harmonic Motion Experiment setup which includes: Supporting system; LHM induction; LHM detection and Complete Setup.

2.2 Transducer Configuration:

The experiments were performed with a Focused Ultrasound (FUS) transducer and an imaging transducer. The FUS transducer was used to induce the harmonic motion inside the sample and the imaging transducer was used to send the diagnostic RF signal to detect the motion.

2.2.1 FUS transducer

The FUS transducer is shown in Figure 12. It had a diameter of 85mm with a central hole of 20mm. FUS transducer operated at a central frequency of 1.1575MHz which was modulated at 75Hz and the focal length is 60mm.

Figure 12. FUS transducer

2.2.2 Diagnostic transducer

A diagnostic ultrasound RF signal was sent by the imaging transducer which is shown in Figure 13. It had diameter of 15.8mm which is smaller than the central hole of the FUS

transducer since it is meant to be placed in that lodging. The focal length is 60mm with a 6dB pass band of 5.28-9.4MHz.

Figure 13. Imaging transducer

2.2.3 Transducers combination

As the estimated response is directly obtained from the tissues undergoing vibration, the information is more dependent on the sample characteristics at the focus than on the surroundings [23]. So both the FUS and imaging transducers have to be focused at the same point (the center of the geometry).

Methods and Materials:

Housing for both of the transducers was designed and built to support and adjust the transducers in order to make them confocal. It was necessary to obtain adjustment in the 3 directions. The housing was made in plastic material (polyetherimide) and brass.

Results:

Figure 14 shows the front view and profile view of the schematic for the housing.
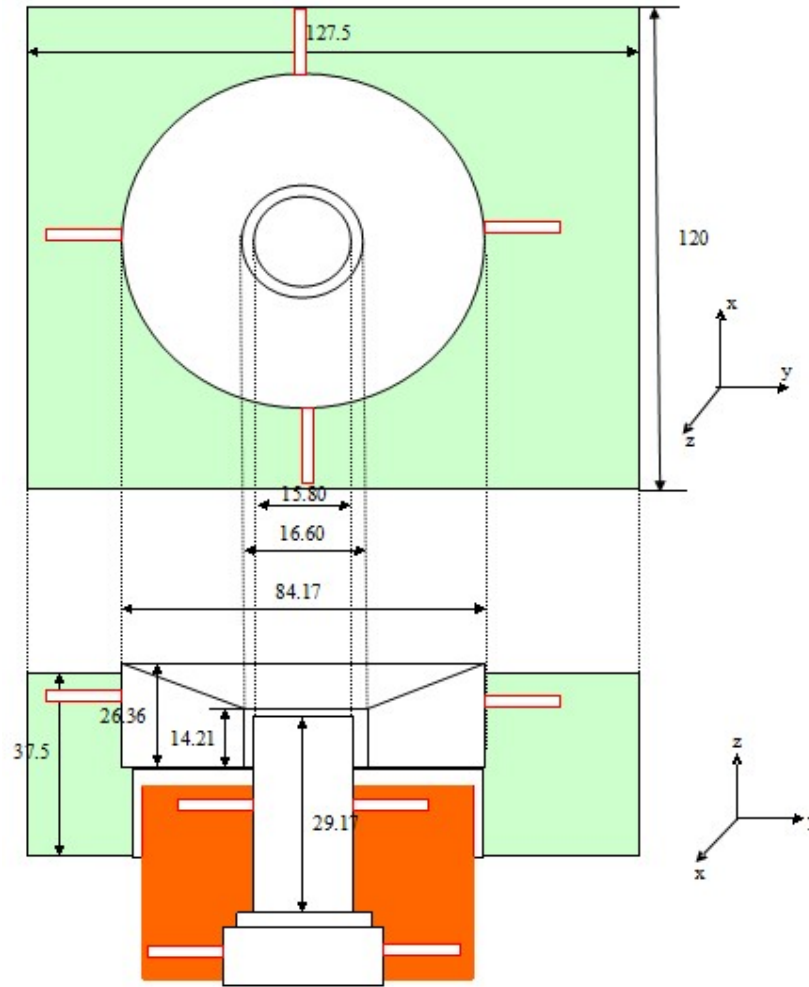
Figure 14. The housing design for the FUS and imaging transducer (measurements are in mm).

The light gray part was the holder for FUS transducer while the dark gray part was the imaging transducer holder. They were connected by threads which allowed for up/down movement of the imaging transducer to adjust the Z direction. The length of the threads was about 7mm since the imaging transducer was small which resulted in limited space for threads. So the adjustment in Z direction could not be only as precise as the threads allowed (1mm pitch threads/mm). The small rectangles in both front view and profile view of the schematic were the screws that allowed moving both the FUS transducer and the imaging transducer in X and Y directions. The screws for adjusting X and Y directions were not designed to be on the same level in order to be able to correct for angle errors. The right side of the FUS transducer holder was designed to be a little bit wider than the left side (7.5mm) in order to connect with an arm for holding the assembly.

Figure 15 shows the front view and side view of the confocal transducers.

<div align="center">front view   side view</div>

Figure 15. The front view and side view of the confocal of FUS transducer and imaging transducer. The focal length for both transducers is 60mm. and inner and outer diameters of the FUS transducer are 20 and 85 mm, respectively. The diameter of imaging transducer is 15.8mm

2.3 Transducer alignment

In order to adjust the FUS transducer and imaging transducer to be confocal we performed pressure measurements with a hydrophone. The objective was to measure the acoustic beam of each transducer to ensure that the focus was at the same point. The imaging transducer was moved until the focuses were aligned.

Methods:

Both of the transducers assembly were mounted on an iron arm and introduced in a water tank. An external hydrophone (0.5mm needle hydrophone, Precision Acoustics, UK) was mounted to a motor and placed under water in front of the assembly. The distance from the hydrophone and both of the transducers was about 60mm which is the focus of both of the transducers. The signal acquired by the hydrophone was first amplified (AG 1020, T&C Power Conversion, New York) and then connected to an oscilloscope (Waverunner 24Xs, LeCroy, New York, USA) and connected with a computer to acquire the pressure information at different positions and then obtain a 2D image of the acoustic pressure field on the XY plane (Figure 16). The difference between the centers should be within an allowed range of 0.5mm.
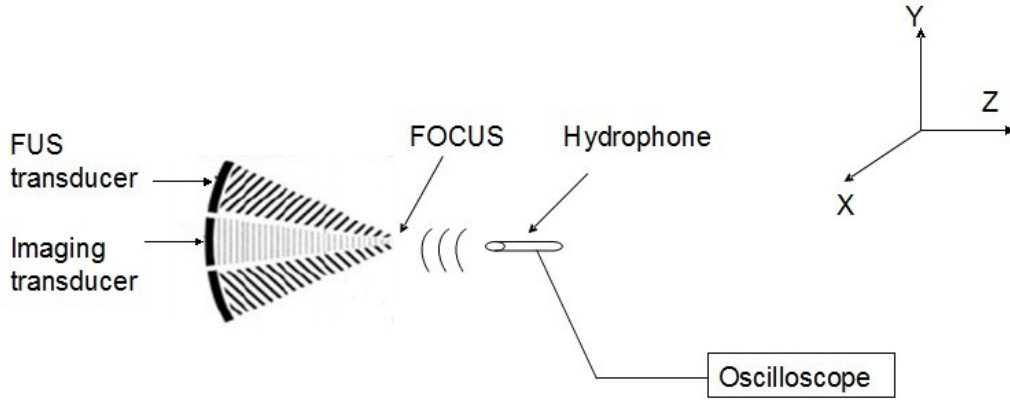
Figure 16. Alignment system

The FUS transducer was excited at its central frequency (1.1575 MHz) with a burst signal generated by two function generators (Agilent 5061 and Agilent 1072, Agilent, USA) and then amplified. The 1072 generated a 10V continuous sinusoidal signal at 1.1575MHz which was gated by the 5061 so 30 bursts were sent every 1 kHz (Figure 17). The gate signal controlled the duty cycle of the continuous sinusoidal signal and also a scanning motor which moved at predetermined locations each time that a gate was received. The reason for the burst was to reduce the total energy sent to the hydrophone, since a continuous wave will damage it. Another advantage is that by using burst we can measure the time of flight and get the distance information. The imaging transducer was excited in the same way. The only difference was that the frequency of its continuous sinusoidal signal was 7.29MHz.
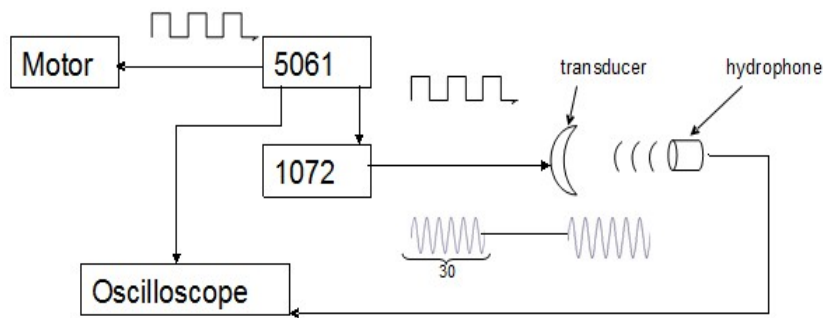
Figure 17. The system connection and the signal control for the alignment experiments

To adjust FUS transducer and imaging transducer, both of the transducers were turned on one at a time and the experiment was conducted under the conditions shown in Table 2.

Temperature:	28.3ºC
Speed of sound in water:	1505m/s
Amplitude of electrical signal:	10Vpp
Signal frequency for FUS transducer:	1.1575MHz
Signal frequency for imaging transducer:	7.29MHz
Number of bursts:	30
Frequency of bursts:	1kHz

Table 2. Condition of alignment experiment

Results:

After adjustment, the hydrophone detected the signal sent by FUS transducer at a time of flight of 40.22μs (equivalent to 60.5mm) while the time of flight of the signal from the imaging transducer was 39.94μs (60.1mm).

In order to form a 2D image of acoustic pressure field, the hydrophone was moved by raster scanning. The field for both transducers was alternatively obtained and after iteration of multiple measurements we can see on Figure 18 and Figure 19 the results of the final pressure field for both FUS transducer and imaging transducer. The maximum pressure for the FUS transducer is at (0, 0) coordinate while the imaging transducer focused at (-0.2,-0.4) which made a difference of 0.44mm which is within the allowed range (0.5mm).
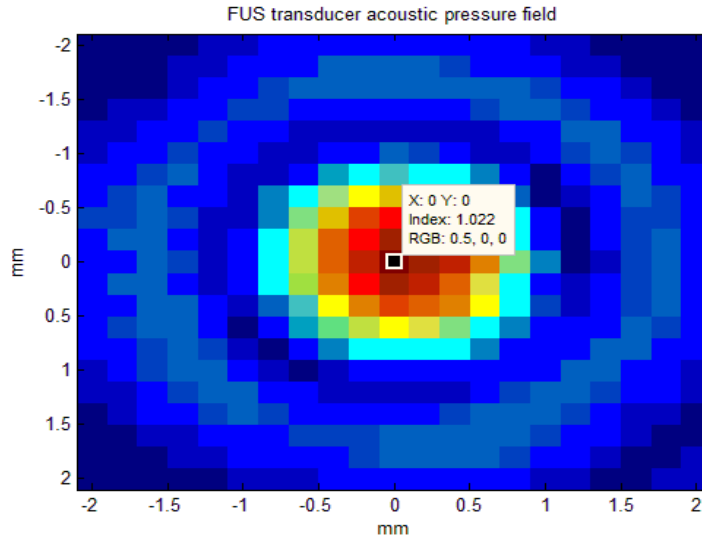
Figure 18. FUS transducer acoustic pressure field with the power centered at (0, 0)
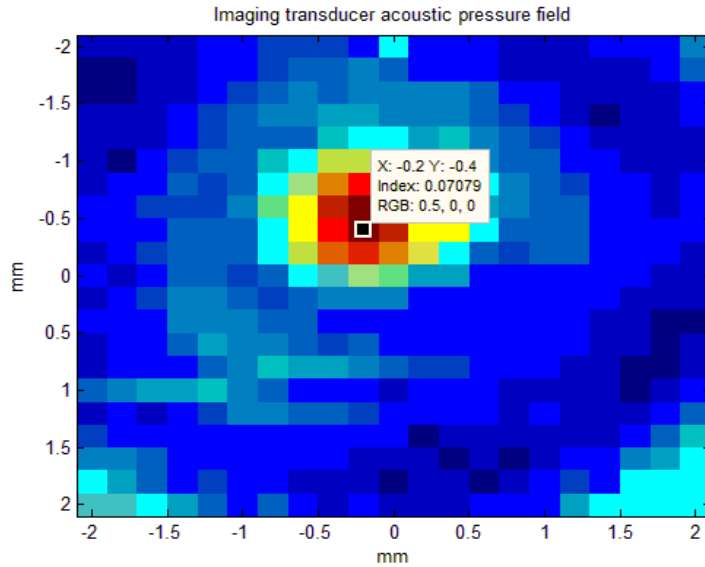
Figure 19. Imaging transducer acoustic pressure field with the power centered at (-0.2,-0.4)

2.4 Local Harmonic Motion Experiment

2.4.1 Supporting system

Since the FUS transducer and imaging transducer are aligned to be confocal. They can be used to detect the localized harmonic motion of the object sample. For the localized harmonic motion experiments, both the transducer assembly was mounted on an arm and introduced in a water tank facing upwards, with the target samples at the top of the tank (Figure 20). In order to form a 2D image of the sample, a positioning slide was used to control the position of transducers manually and rulers were installed at the x-axis and y-axis of the positioning slide to read the coordinates of the transducer.

The big challenge for this step was how to make the transducers leveled since both of them are heavy. The housing also increased the weight of the left side of supporting system which made it even harder to keep balance. In order to solve this problem, we added another arm on the top and a big piece of wood under the positioning slide. Two clamps were also added to prevent the system from leaning leftwards.
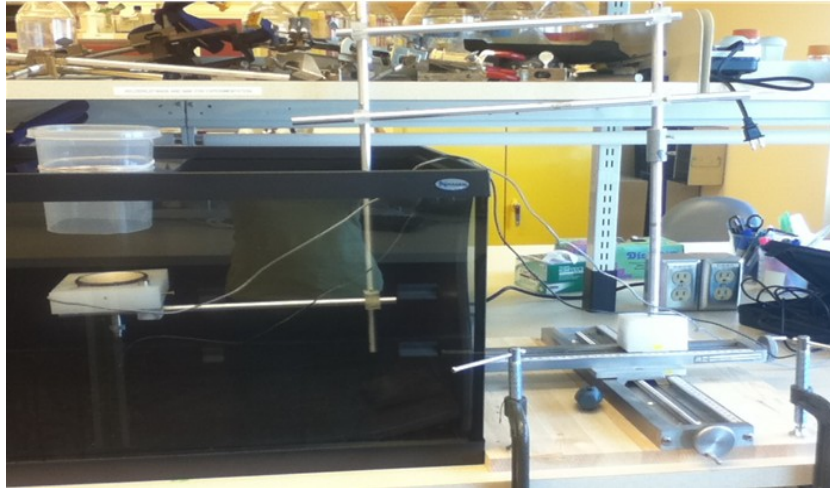
Figure 20. Supporting system setup

2.4.2 LHM induction

The time varying radiation force can be applied by overlapping two ultrasound beams radiating at slightly different frequencies or by using an amplitude-modulated single-frequency ultrasound beam. We chose to use a modulated ultrasound beam. For this, the Focused Ultrasound (FUS) was excited by an amplified modulated signal and the induced oscillation of the tissue is generated at the modulation frequency of the excitation.

Method and Materials:

The modulated signal (Figure 22) was generated by a generator card CG4302 (CompuGen4302: A 300MHz, 4-channel, 12-bit digital to analog waveform generator card for the PCI bus, Gage, USA) and then amplified (AG 1020, T&C Power Conversion, New York). The modulation signal contained 7 cycles, including a first and last cycle without excitation and 5 cycles of modulated signal in the middle. One key point for this step is that CG4302 is not compatible with the windows 7 operating system. So the card was installed in another computer with windows XP system and controlled remotely by the master computer (Figure 21).
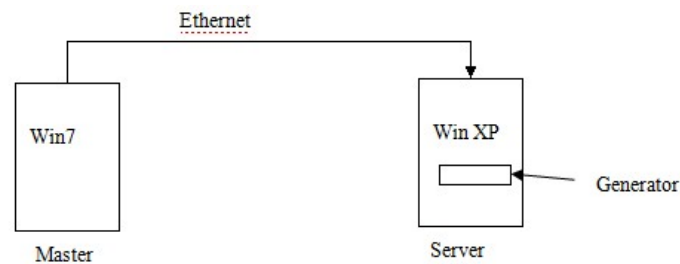
Figure 21. The connection schematic of Win7 and WinXp computers

We used C++ programming software to send commands to the server computer. The

commands were passed through Ethernet. The server computer was always in listening mode. Once it received the commands from master computer, it will program the generator cards as required and sent the waveform as needed.

Results:

Figure 22 shows the modulated waveform sent from the generation card. The 1.1575MHz FUS was modulated at 75Hz. It had 7 cycles, the first and last cycle with no excitation and 5 middles cycles of modulated signal. This signal was acquired from the output of the generator card using a digitizer card (ATS460:Waveform digitizer with two 14-bit resolution analog input channels and sampling rate of 125 MS/s and 8 million samples in memory, Alazartech, Montreal, ON, Canada).
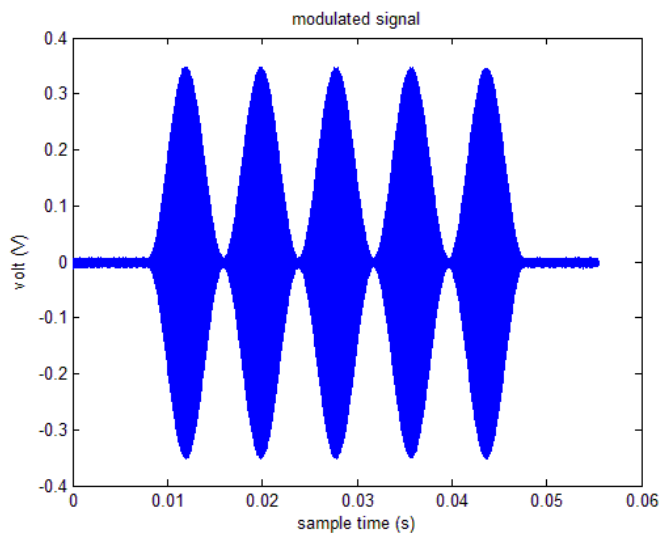
Figure 22. Modulated signal at a central frequency of 1.1575MHz and modulation frequency of 75Hz

2.4.3 LHM detection

The imaging transducer detects the motion by monitoring the excited tissues using an ultrasound echo-graphic system. We sent an ultrasound pulse, and obtained the echoes coming back from the target sample/tissues, and by analyzing the changes in these echoes we can determine the behavior of the target sample/tissues.

Method and Materials:

To send the ultrasound pulse we excited the imaging transducer using a pulser/receiver (DPR300 Ultrasonic Pulser/Receiver, JSB, NY, USA) and we detected the echoes with the pulser/receiver. Since the frequencies obtained on the pulser/receiver are typically between 20 kHz-10MHz, we usually call this signal a radio frequency (RF) signal. The received RF signal was acquired during the LHM. Pulses were sent during the radiation force application

at a pulse repetition frequency (PRF) of 5 kHz to detect the localized harmonic motion of the sample as a function of time.

Results:

Figure 23 shows a single RF signal obtained from a silicon sample. We can see a big peak at the first interface of the sample followed by a smaller peak and another big peak at 77µs and 80µs respectively. The big peak at 80µs was the echo coming back from the target sample while the smaller one in front of it was the echo from a plastic bag used to support the sample.
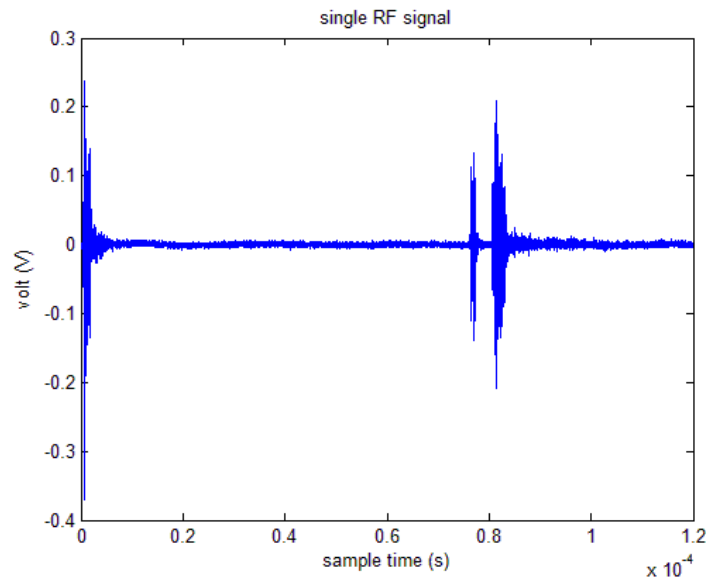
Figure 23. Radiation Force signals

Then the data was acquired using a PCI digitizer card (ATS460) and stored on a hard drive. We sent 462 RF signals for each acquisition and compared each adjacent 2 RF signals to get the information of displacements (the details will be talked about in Chapter 5).

2.4.4 Complete Setup

All the individual parts previously described are put together as shown in Figure 24 and Figure 25.

Figure 24 shows the schematic of the complete setup: The output 1 of generator card (CG4302, Gage, USA) was connected to the amplifier and then to the FUS transducer. So the FUS transducer sent modulated signals generated by the generator to the target sample. The other output of the generator card was a marker which was used as the trigger for the acquisition. It was therefore connected to the external trigger input of the digitizer card (ATS

460). And the trigger output of the digitizer card sent triggers to the pulser/receiver to synchronize the triggers. The pulser/receiver sent the detective RF signal followed the trigger to the imaging transducer to track localized harmonic motion. The imaging transducer acquired the signals reflected back from target sample and passed it through a limiter, small signal amplifier and a filter and then to channel A of the digitizer card. This data was finally stored on the computer.

Figure 25 shows the actual experiment setup.
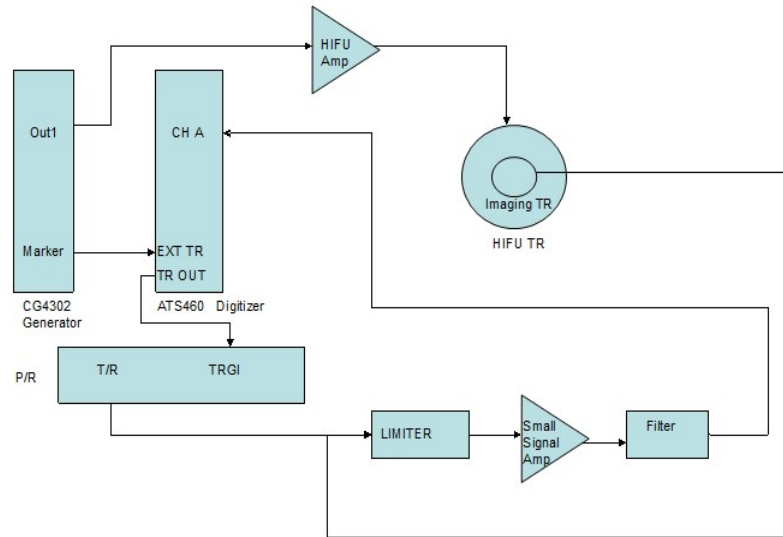
Figure 24. The schematic of the complete setup
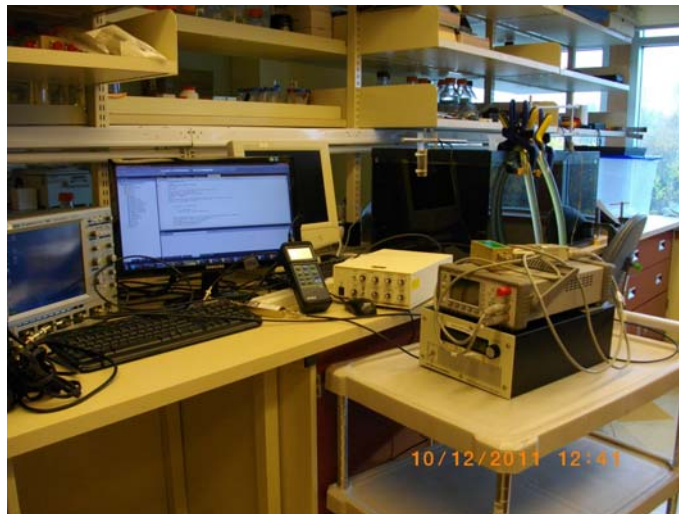
Figure 25. Actual experiment setup

3. Analog Signal Filter Design

3.1 Introduction:

On the schematic of setup shown in Figure 24, one of the most critical stages is the acquisition of the signal, since signals acquired by the imaging transducer always contain different sources of noise such as acoustic (standing waves, reflection from the tank, mechanical cross-talk between the transducers) and electric (electric cross-talk, noise of the high power amplifier). The main source of noise came at the harmonic frequencies of the FUS transducer system. So in order to get rid of the noise, electronic filters need to be designed and implemented into our LHM experiment system.

In circuit theory, a filter is an electrical network that alters the amplitude and/or phase characteristics of a signal with respect to frequency. Ideally, a filter will not add new frequencies to the input signal, nor will it change the component frequencies of that signal, but it will change the relative amplitudes of the various frequency components and/or their phase relationships. Filters are often used in electronic systems to emphasize signals in certain frequency ranges and reject signals in other frequency ranges. Such a filter has a gain which is dependent on signal frequency [31].

This chapter will describe analog filters we designed and implemented into the Localized Harmonic Motion (LHM) experiment and their testing results in the experiments.

3.2 Analog Filters

3.2.1 Expected Frequency Spectrum

The imaging transducer we used had a 6dB pass band of 5.28-9.3 MHz. So in order to eliminate noise out of the pass band, a band-pass filter had to be designed and built which could help us get rid of the low and high frequency noise. So we decided to build a band-pass filter between 4.7 and 10.7MHz.

Besides the low and high frequency noise, the FUS transducer also caused noise peaks at the harmonic frequencies of the transducer. As we know the fundamental frequency FUS transducer is 1.1575MHz and the harmonics will be at multiples of the 1.1575MHz. Since the band pass filter was supposed to filter the frequencies out of our imaging bandwidth (5.28 to 9.3 MHz), only the 5th (5.79MHz), 6th (6.95MHz), 7th (8.10MHz), 8th (9.26MHz), and 9th (10.42MHz) harmonics were still a concern. So five notch filters were designed in order to clean the signal.

3.2.2 Filter Topology Selection

There are several choices of filters such as Butterworth, Chebyshev or Bessel so that the filter topology needs to be selected first before building or implementing the filters [32].

Here is the comparison of several filter topologies:

The <u>Butterworth</u> filter is the best compromise between attenuation and phase response. It has no ripple in the pass-band or the stop-band and because of this is sometimes called a maximally flat filter. The Butterworth filter achieves its flatness at the expense of a relatively wide transition region from pass-band to stop-band, with average transient characteristics. The values of the elements of the Butterworth filter are more practical and less critical than many other filter types.

The <u>Chebyshev</u> filter has a smaller transition region than the same order Butterworth filter, at the expense of ripples in its pass-band. This filter gets its name from the Chebyshev criterion, which minimizes the height of the maximum ripple.

The <u>Bessel</u> filter is optimized to obtain better transient response due to a linear phase (i.e. constant delay) in the pass-band. This means that there will be relatively poorer frequency response (less amplitude discrimination) [33].

We decided to implement a Butterworth filter, which is a classic filter approximation that has a smooth response in both the pass-band and stop-band. It provides the most linear phase response, which was critical to avoid miscalculations on the cross-correlation. However, a Butterworth filter will require a much higher order to match the transition band characteristics of a Chebyshev filter [32]. The Chebyshev filters improve on the amplitude response at the expense of transient behavior. The Bessel approximation provides better phase characteristics, but has very poor transition band characteristics.

So considering that since we wanted to get rid of the unwanted frequencies outside the pass-band, we needed a band-pass filter with a good attenuation at the stop-band. And also, in order to get correct value of time delay for cross-correlation calculations which correspond to the LHM displacements information, the linear phase response of the filters was crucial to the LHM experiment. The Butterworth filter had the best compromise between attenuation and phase response.

3.2.3 Filters Type Design

Analog filters can be divided into passive and active filters.

The Filters which are made up of passive components: resistors (R), capacitors (C), and inductors (L) are referred to as passive filters. While Active filters use amplifying elements,

especially op amps, with resistors and capacitors in their feedback loops, to synthesize the desired filter characteristics [31].

By comparing characteristics of passive filters and active filters, the passive filters showed several advantages over active ones as summarized below:

Since a passive filter is simply a filter that uses no amplifying elements (transistors, operational amplifiers, etc.), it is the simplest (in terms of the number of necessary components) implementation of a given transfer function. Passive filters have other advantages as well. Because they have no active components, passive filters require no power supplies. Since they are not restricted by the bandwidth limitations of op-amps, they can work well at the high frequency needed in our LHM experiment (MHz). They can be used in applications involving larger current or voltage levels than can be handled by active devices. Passive filters also generate little noise when compared with circuits using active gain elements. The noise that they produce is simply the thermal noise from the resistive components and, with careful design, the amplitude of this noise can be very low [31].

All passive filters have been designed by the software named AADE Filter Design and Analysis.exe (Version 4.5, http://www.aade.com/filter.htm) and built in-house

3.2.4 Filter Implementation

Two types of filters were built and implemented into LHM experiment:

(1). Band-pass filters (high-pass filter and low-pass filter)

(2). Notch filters (5th, 6th, 7th, 8th, 9th)

3.2.4.1 Band-pass Filter (4-10MHz)

A band-pass filter enables a band of frequencies to be passed through, while those frequencies outside the pass band of the filter are rejected. The higher the filter order is the closer to the ideal the response will be. But the cost of the filter (time delay, number of elements, cost, power consumption, etc.) will increase as well [32]. So considering the trade-off and trying the software of different order of filters, we decided to build a 6th order Butterworth band-pass filter (4-10MHz band pass). According to the expected frequency response (Figure 26), it shows good attenuation in the stop-band and low level insertion loss within pass-band. And also the soldering work for 6th order Butterworth filter is not complicated either (Figure 27).
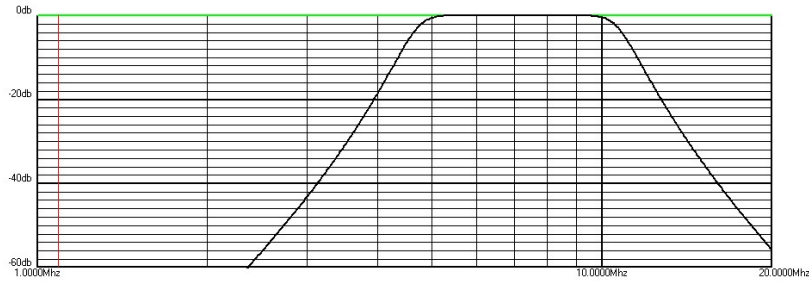
Figure 26. Expected frequency response of 6th order Butterworth filter (obtained with AADE Filter Design and Analysis.exe)

Methods and materials:

Figure 27 below illustrates the circuit of the 6th-order Butterworth band-pass filter which was implemented. Inductors blocked the high-frequency signals and conducted low-frequency signals, while capacitors did the reverse.
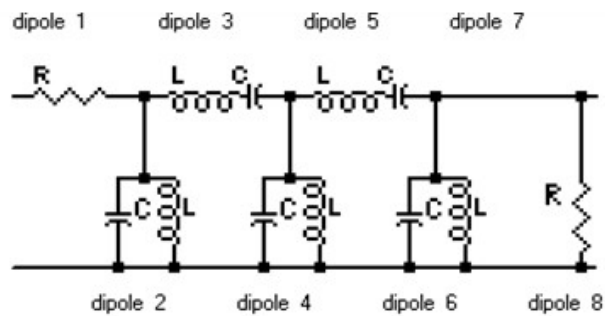
Figure 27. 6th-order Butterworth band-pass filter schematic

Dipole1: R1=50Ω.   Dipole2: C2=390pF, L2=2.2μH.

Dipole3: C3=270pF, L3=3.3μH.   Dipole4: C4=0.0012uF, L4=0.47μH.

Dipole5: C5=270pF, L5=3.3μH.   Dipole6: C6=390pF, L6=2.2μH.

Dipole8: R8=40Ω.

Results:

Figure 28 and Table 3 show the testing result of the band-pass filter.

Pass Band(-20dB):	3.6MHz
Stop Band(-20dB):	10.4MHz
Average Insertion Loss:	-6.5dB

Table 3. Testing result of band-pass filter obtained with a network analyzer (HP8751A, Agilent, USA)

As we can see the insertion loss in the pass band was -6.5dB which was too high for the LHM imaging since the signal is only about 1mV. The reason is that the order for the filter is too high and the higher the order is, the higher the insertion loss is. So this passive band-pass filter was not adequate for the experiment and needed to be modified.
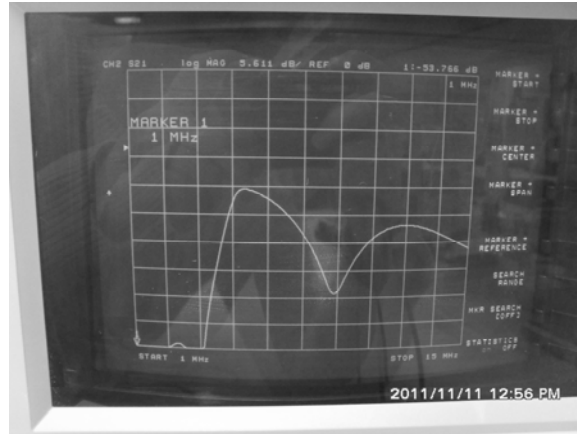
Figure 28. Testing result of frequency spectrum for Butterworth band pass filter obtained with a network analyzer (HP8751A, Agilent, USA)

Modification:

We broke this band-pass filter into two parts: a low-pass filter (BLP-10.7+, Minicircuits, USA) combined with a high-pass filter which made it easier to control.

For the low-pass filter, we took a commercial filter with a pass band at 13.14MHz. Figure 29 and Table 4 show the testing result of this low-pass filter.

Pass Band(-3dB):	13.14MHz
Stop Band(-20dB):	16.25MHz
Average Insertion Loss:	-0.5dB

Table 4. Testing result for Low Pass filter
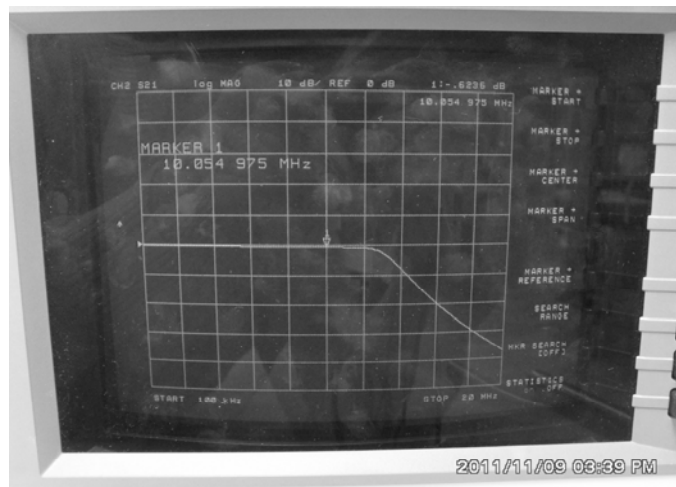
Figure 29. Testing result of frequency spectrum for commercial low-pass filter

Methods and materials:

For the high-pass, we built a filter with a stop band of 4MHz. And also we changed to a circuit board with ground plane to improve its performance. This structure is a layer of

copper that appears to most signals as an infinite ground potential. This helps reduce noise and helps ensure that all integrated circuits within a system compare different signals' voltages to the same reference potential. And also we changed the filter order 6 into order 5 as it is known that the higher the order is, the higher the insertion loss is. Figure 30 below shows the circuit of the high-pass filter.
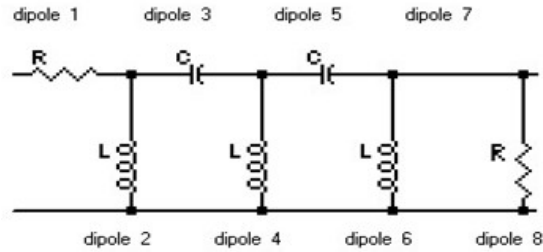
Figure 30. 5th-order Butterworth high-pass filter schematic

Dipole 1: R1=50Ω.   Dipole 2: L2=3.3uH.

Dipole 3: C3=500pF.   Dipole 4: L4=1uH.

Dipole 5: C5=500pF.   Dipole 6: L6=3.3uH.

Dipole 8: R8=50Ω.

Results:

Figure 31 and Table 5 display the testing results of the High-pass filter. The average insertion loss of the high-pass filter is -0.8dB.
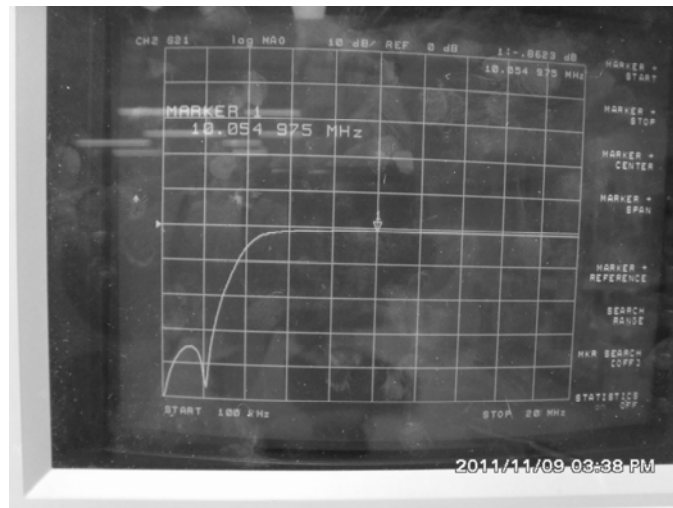
Figure 31. Testing result for High-pass filter

Start Band(-3dB):	4.7MHz
Stop Band(-20dB):	2.9MHz
Average Insertion Loss:	-0.8dB

Table 5. Testing result for High Pass filter

3.2.4.2 Notch Filters

A notch filter is a band-stop filter with a narrow stop-band used to remove a specific interfering frequency. In localized harmonic motion imaging, the frequencies that have to be notched are only the 5th (5.79MHz), 6th (6.95MHz), 7th (8.10MHz), 8th (9.26MHz), and 9th (10.42MHz) harmonics as it has been explained above.

Methods and materials:

Figure 32 shows the schematic of 5th order Butterworth notch filter used for all the frequencies to be notched.
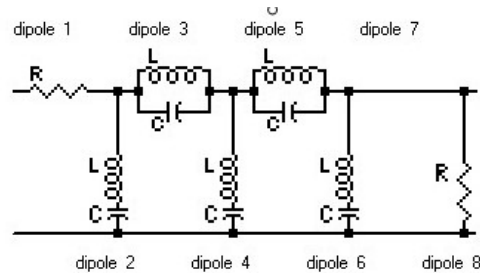
Figure 32. 5th order Butterworth notch filter schematic

Results:

Table 6 below illustrates the testing result for the five notch filters (including the attenuation at harmonic frequencies and the maximum attenuation of each filter). The insertion loss of these five notch filters was always below -3dB, so the bandwidth of the filters is not shown here. The insertion loss for all these notch filters was also too large for the Localized Harmonic experiment for the same reason as the first band-pass filter, the order of these notch filters was too high.

And also, by comparing the harmonic frequencies of each notch filters and the frequencies at which it achieved the maximum attenuation, the deviation was also too big for most of the notch filters (like 7th and 8th harmonic notch filters). This was because the elements we used were all through-hole components which could not give us a precise value. So we needed to modify these notch filters too.

Harmonics	5th	6th	7th	8th	9th
Frequency	5.79MHz	6.95MHz	8.10MHz	9.26MHz	10.42MHz
Attenuation	-11.16dB	-10.94dB	-12.10dB	-4.50dB	-29.26dB
Maximum Attenuation	-29.26dB (@6.4MHz)	-24.19dB (@7.6MHz)	-31.09dB (@9.3MHz)	-22.63dB (@10.2MHz)	-29.49dB(@10.4MHz)
Average Insertion Loss	-4.10dB	-4.15dB	-5.32dB	-6.30dB	-3.55dB

Table 6. Testing results of 5 notch filters obtained using a network analyzer (HP8751A, Agilent, USA)

Modification:

Figure 33 shows the modified schematic for the new notch filters. First, in order to reduce the insertion loss of the circuits, we decreased the order of all the notch filters from 5 to 3. And also, to decrease the deviation between the harmonic frequencies and the maximum attenuation frequencies, we replaced all the through-hole elements with surface mount elements since they have advantages such as:

(1). Lower resistance and inductance at the connection (leading to better performance for high frequency parts).

(2). Better mechanical performance under shake and vibration conditions.

(3). Fewer unwanted RF signal effects in surface mount parts when compared to leaded parts, yielding better predictability of component characteristics.
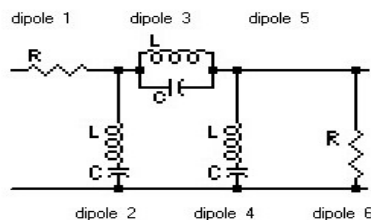
Figure 33. 3rd order Butterworth notch filter schematic

Table 7 shows the testing results of the 3rd order Butterworth notch filters from attenuation, maximum attenuation, bandwidth and average insertion loss. The attenuation of these 5 notch filters all achieved a low level and the deviation between the 5 harmonic frequencies and the maximum attenuation frequencies also decreased to an acceptable range.

One thing needs to be mentioned here is that for the 6th and 9th harmonic notch filters, both the attenuation and the bandwidth were larger than the other three notch filters (5th, 7th and 9th harmonic notch filters). The reason for this was that the inductors we needed for the

6th and 9th harmonic notch filters were much smaller than the range of the surface mount elements we could obtain so that we had no other choice but to use through-hole elements which led to the larger bandwidth and attenuation.

Harmonics	5th	6th	7th	8th	9th
	5.79MHz	6.95MHz	8.10MHz	9.26MHz	10.42MHz
Attenuation	-8.9dB	-16dB	-3.4dB	-6.37dB	-21dB
Maximum Attenuation	-10.2dB (@5.9MHz)	-31dB (@6.3MHz)	-7.83dB (@8.7MHz)	-6.9dB (@9.4MHz)	-39dB (@10MHz)
Bandwidth	0.96MHz	2.53MHz	1.33MHz	1.04MHz	2.21MHz
Average Insertion Loss	-0.35dB	-0.50dB	-0.30dB	-0.40dB	-0.50dB

Table 7. Testing result for five 3rd order Butterworth notch filters.

3.2.5 Testing Results in Localized Harmonic Motion Experiments

After building all the filters (including the low-pass filter, high-pass filter and all the 5 notch filters), we included them in the Localized harmonic motion experiment system for testing. In order to evaluate the impact of each filter, we introduced the filters one by one. The order in which we added the filters was: 1) the low-pass filter, 2) the high-pass filter and 3) the 5 notch filters starting from the 5th to the 9th harmonic.

Since the main noise source of the system was the harmonics brought by the FUS transducer system, two groups of data were acquired for comparison. The first group of data was acquired with the FUS transducer off to get a clean signal and frequency spectrum while the other group was acquired with the FUS transducer on in order to have an overall look at the noise reduction.

Figure 34 shows the testing results of signals without FUS noise (focus part from 70µs to 95µs) and Figure 35 shows the testing results of the signals with the FUS noise but without any filters. These two testing results will be used as references.
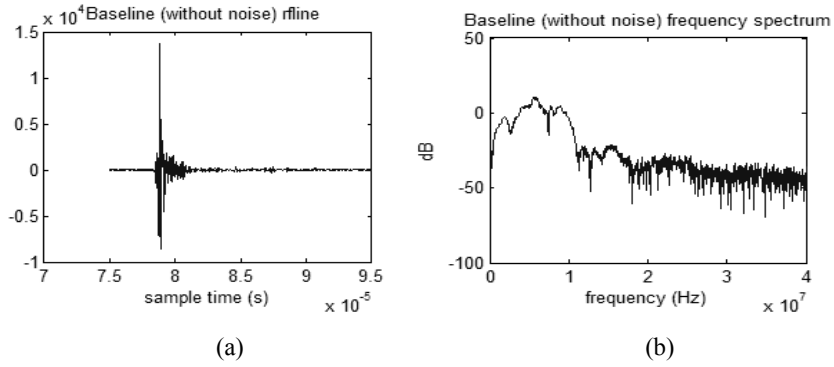
Figure 34. signal (a) and frequency spectrum (b) with the FUS transducer off
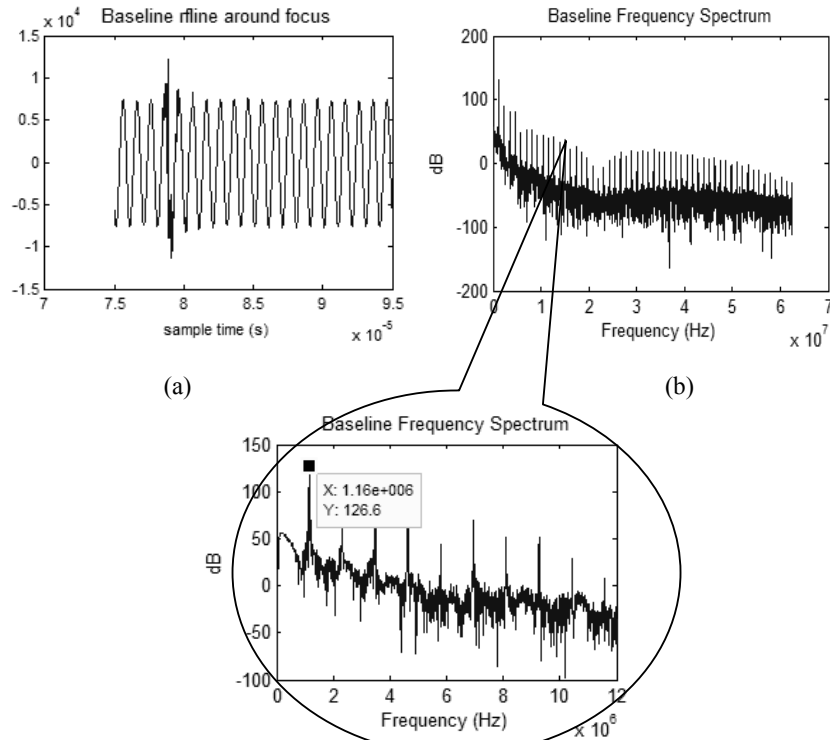
Figure 35. signal (a) and frequency spectrum (b) with the FUS transducer on

Comparing Figure 34(a), (b) with Figure 35(a),(b), the signals and its frequency spectrum in Figure 35(a),(b) contains large amount of noise at the harmonic frequencies brought by the FUS transducer. According to the zoomed in plot of the frequency spectrum (Figure 35(b)), we can see the frequency value of the first noise peak was at 1.16 MHz which was the central frequency for FUS transducer (1.1575MHz).

Due to the high level of noise, we started adding the filters we built into LHM experiment system and analyzed the response. The purpose of building filters is to the make the signal as close as possible to the signal frequency spectrum without FUS noise.

Testing Results with only Low-pass Filter:

Figure 36 below shows the testing result of the signals only with the low-pass filter and its frequency spectrum. Since the stop band of the low pass filter was 16.25MHz, the harmonic noises which were higher than this value disappeared. It proved the low pass filter worked very well.

(a)   (b)
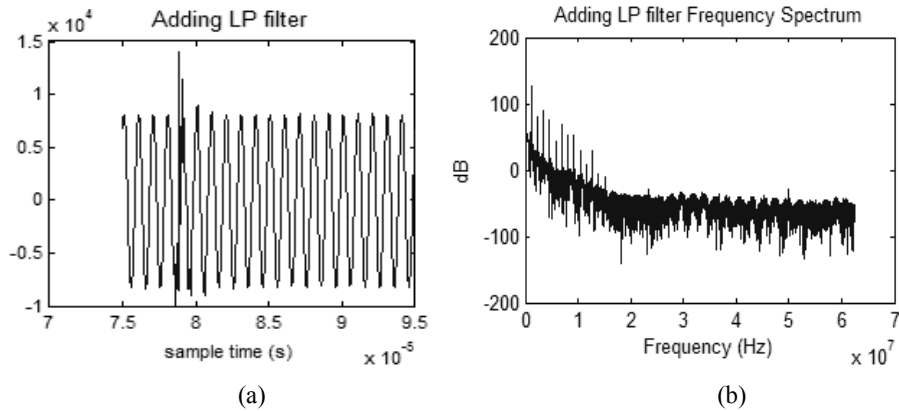
Figure 36. (a) is the signal with the FUS transducer on and only with the low-pass filter and (b) is the frequency

spectrum of the signal shown (a)

Testing Results with Low-pass Filter and High-pass Filter:

(a)   (b)
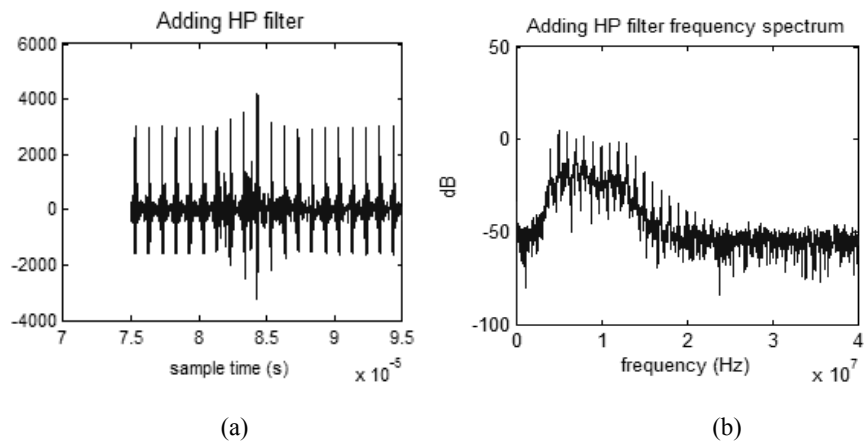
Figure 37. (a) is the signal with the FUS transducer on and with both low-pass filter and high-pass filter and (b)

is the frequency spectrum of the signal show (a)

After adding the high-pass filter into the LHM experiment system, the central frequency noise peak and other low frequency harmonic peaks disappeared as expected since it could not pass through the high-pass filter with the start band at 4.71MHz (Figure 37).

Testing Results with Notch Filters:

Then we added all the notch filters into the LHM experiment system as well and got the testing results below. Figure 38 (a) shows the signal around the focus after all the filters were

implemented while Figure 38 (b) shows the frequency spectrum. Since the 6th harmonic filter had a large bandwidth and attenuation, so there is a big notch at around 6.95MHz (6th harmonic frequency) in the frequency spectrum. This may deform the signal too much so that we decided to remove the 6th harmonic notch filter and still obtained a satisfactory signal.

(a)   (b)
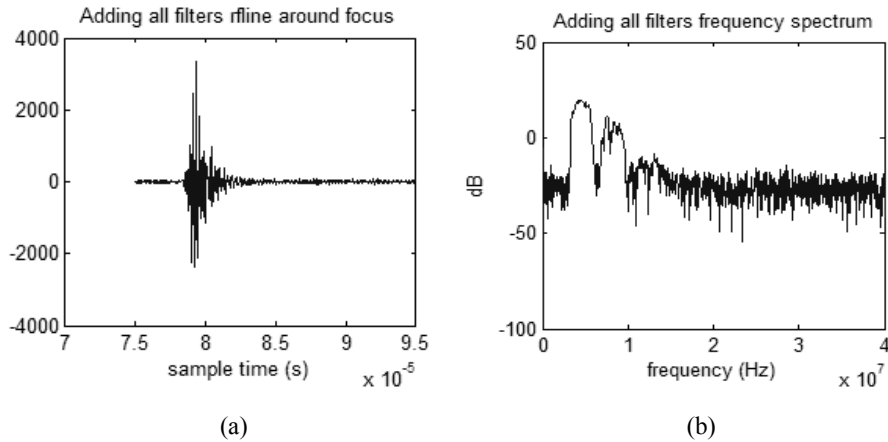
Figure 38. (a) is the signal with the FUS transducer on and with all the filters expect the 6th harmonic notch filter and (b) is the frequency spectrum of the signal shown in (a)

After the 6th harmonic notch filter was removed, we got the final testing results of filters (Figure 39 (a) and (b)). Comparing with the signal without FUS noise (Figure 34 (a) and (b)), the signal below seems nice and clean and the frequency spectrum also looks acceptable.

(a)   (b)
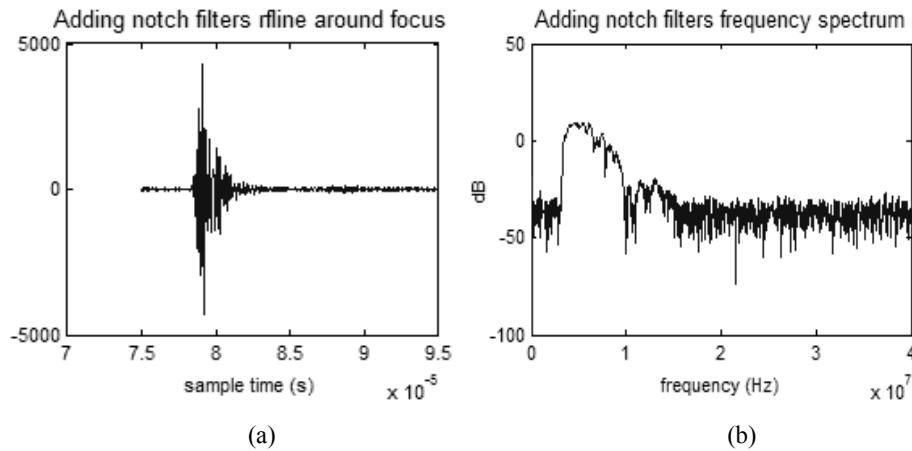
Figure 39. (a) is the signal with the FUS transducer on and with all the filters expect the 6th harmonic notch filter and (b) is the frequency spectrum of the signal shown in (a)

3.2.6 Discussion:

3.2.6.1 9th Harmonic Notch Filter

Both of the 6th harmonic notch filter and 9th harmonic notch filter have a large bandwidth

and attenuation which was caused by the through-hole elements. But we only saw a big notch at the frequency at around 6.95MHz (6th harmonic frequency) (Figure 38 (a) and (b)). The reason for this is that even if the 9th harmonic notch filter has a big bandwidth and attenuation, the higher frequency side of the notch filter was already beyond the pass band of the imaging transducer acquisition system (4.7MHz-10.7MHz) so that we couldn't see the big notch here in the frequency spectrum (Figure 40). However, if you compare Figure 40(a) with (b), you can still find small difference inside the circle. The frequency spectrum with all notch filters (except the 6th harmonic notch filter) seems to be steeper than the one only with 5th harmonic notch filter.

(a)   (b)
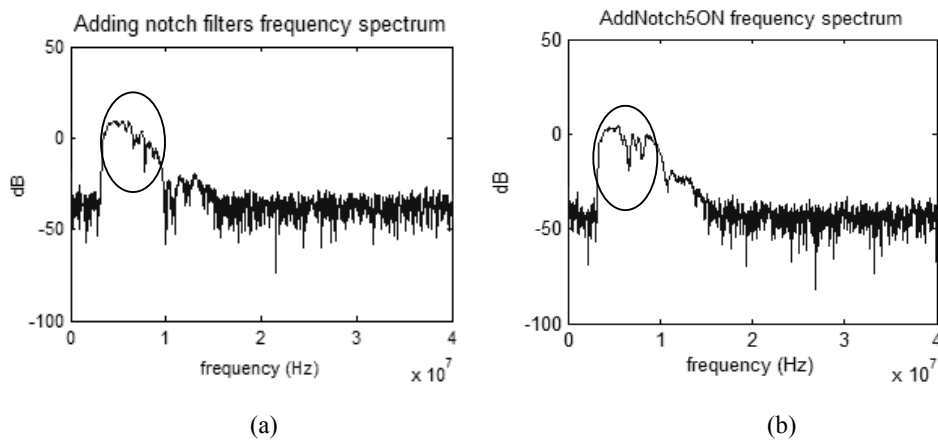
Figure 40. (a)The frequency spectrum of the signal with the FUS transducer on and with all the filters except the 6th harmonic notch filter. (b) The frequency spectrum of the signal with the FUS transducer on and with high-pass, low-pass and 5th harmonic notch filter.

3.2.6.2 Inductors in Passive Filters

In Table 7 we noticed that for the 6th and 9th harmonic notch filters, both the attenuation and the bandwidth were much larger than the other three notch filters (5th, 7th and 9th harmonic notch filters) which resulted from the through-hole inductors as we talked before.

Inductors are usually necessary for the synthesis of most useful passive filter characteristics, and these can be prohibitively expensive if high accuracy (1% or 2%, for example), small physical size, or large value is required. In this case, the inductors we used were 10% of the arbitrary value which led to the big change into the circuits. Furthermore, we even tried to build the inductors by ourselves with tubes and wires. However, tuning inductors to the required values is time-consuming and expensive when producing large quantities of filters.

4. Digital Signal Filter Design

4.1 Introduction:

After building all the analog filters, we conducted a series of LHM experiments. However, we were not able to obtain the displacements consistently. Considered that this can still come from noise remaining in the signal, we decided to implement digital filters into LHM data processing part.

Digital filter is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. In this section, we applied several kinds of notch filters for harmonic noises and conduct further analysis on the obtained LHM displacements.

Digital filters are usually classified by the duration of their impulse response, which can be either finite or infinite. The methods for designing and implementing these two filters classes differ considerably. Finite impulse response (FIR) filters are digital filters whose response to a unit impulse (unit sample functions) is finite in duration. This is in contrast to infinite impulse response (IIR) filters whose response to a unit impulse (unit sample function) is infinite in duration. FIR and IIR filters each have advantages and disadvantages, and neither is best in all situations [34]. This chapter will talk about both IIR and FIR notch filters we implemented in the LHM experiments.

4.2 IIR notch filters

The general form for an infinite impulse response (IIR) filter's output $y[k]$ at time k is given by

$$y[k] = b_0 x[k] + b_1 x[k-1] + ... + b_M x[k-M] - a_1 y[k-1] - a_2 y[k-2] - ... - a_N y[k-N] \quad (4.2.1)$$

Where:

M is the feed forward filter order;

b_i are the feed forward filter coefficients;

N is the feedback filter order.

a_i are the feedback filter coefficients.

$x[k]$ is the input signal;

$y[k]$ is the output signal.

This equation indicates that the filter's output is a linear combination of present input, the M previous inputs, and the N previous outputs. A direct realization of Eq. (4.2.1) is shown in Figure 41 using the signal flow graph notation [34].
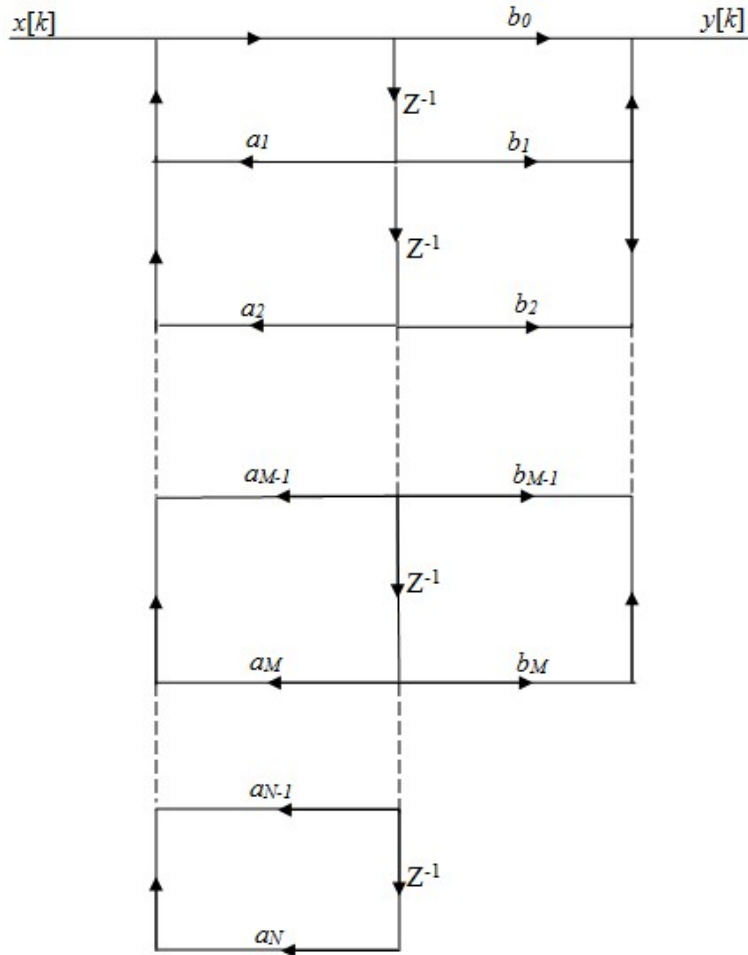
Figure 41. Signal flow graph for an IIR filter system [34].

Methods:

As we talked in Chapter 3.2.2, since the Butterworth filter is the best compromise between attenuation and phase response, we still chose Butterworth topology for digital filter design.

Figure 42 (a) shows the raw RF signal we acquired with FUS noise and (b) shows the RF signal without FUS noise as reference. Figure 43 (a) and (b) shows the frequency spectrum of RF signals with and without FUS noise. As we can see FUS noise had a big effect on the acquired RF signals. For the reason that the higher the order of filter is, the more the time it consumes, after some adjustment, we decided to apply a 2nd order Butterworth IIR notch filter

(see Appendix 1.1) with the bandwidth of 0.03MHz to the acquired data. It will be our best choice if a low order filter could satisfy our requirement.

(a)   (b)
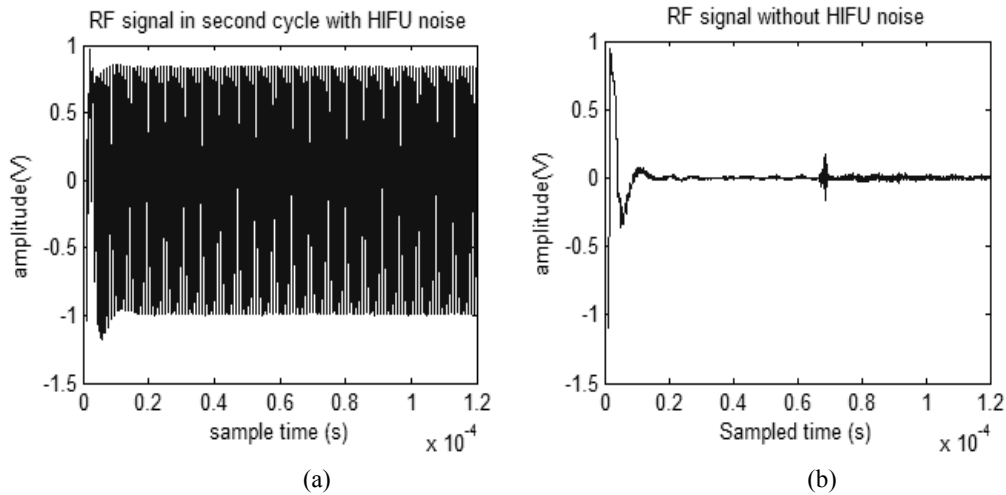
Figure 42. RF signal with and without FUS noise. (a) is the RF signal with FUS noise. (b) is the RF signal without FUS noise

(a)   (b)
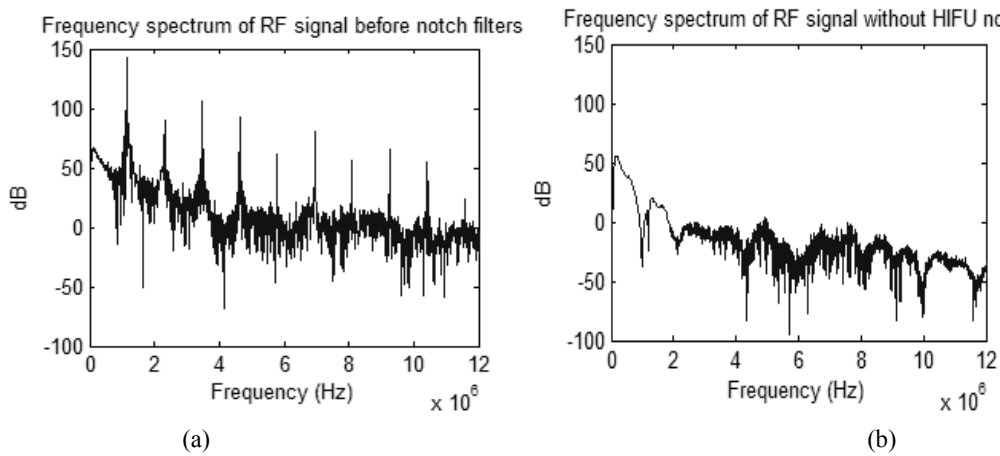
Figure 43. Frequency spectrum of RF signals with and without FUS noise. (a) is the RF signal with FUS noise. (b) is the RF signal without FUS noise

Results:

Figure 44 shows the RF signal with FUS noise processed by the IIR Butterworth filter. The amount of noise was reduced notably. However, the beginning part still contained noise (within 40μs) but this part was far away from the focus (80μs) which was not significant.

Figure 45 indicates the frequency spectrum of RF signal processed by the IIR filter. After comparing with Figure 43, we noticed that the value of all the central and harmonic frequency peaks decreased significantly after being processed by the IIR notch filter.
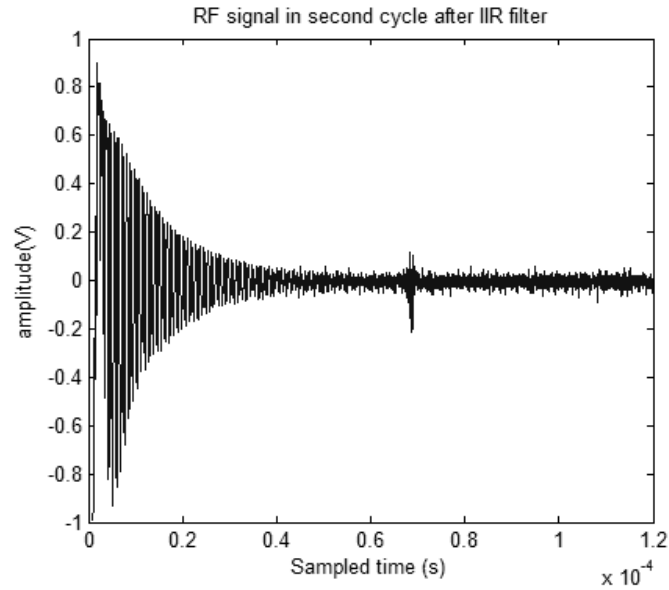
Figure 44. RF signal with FUS noise processed by the IIR Butterworth filter
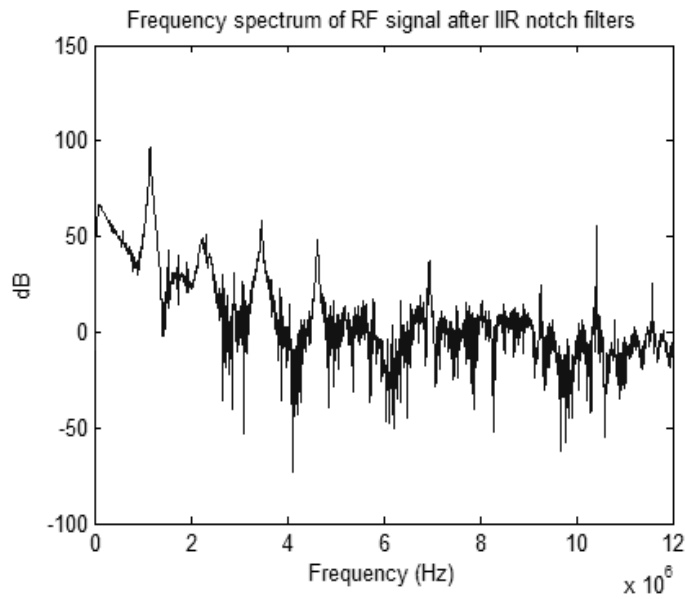
Figure 45. Frequency spectrum of RF signal with FUS noise processed by the IIR filter.

4.3 FIR Notch Filter

4.3.1 Regular FIR Notch Filter

The general form for a finite impulse response (FIR) filter's output $y[k]$ at time k is given by

$$y[k] = b_0 x[k] + b_1 x[k-1] + ... + b_M x[k-M] \tag{4.3.1}$$

Where:

b_i are the filter coefficients, also known as tap weights, that make up the impulse

response;

$x[k]$ is the input signal;

$y[k]$ is the output signal;

M is the filter order.

This equation indicates that the FIR filter's output is a linear combination of M previous inputs. A direct realization of Eq. (4.3.1) is shown in Figure 46 using the signal flow graph notation.
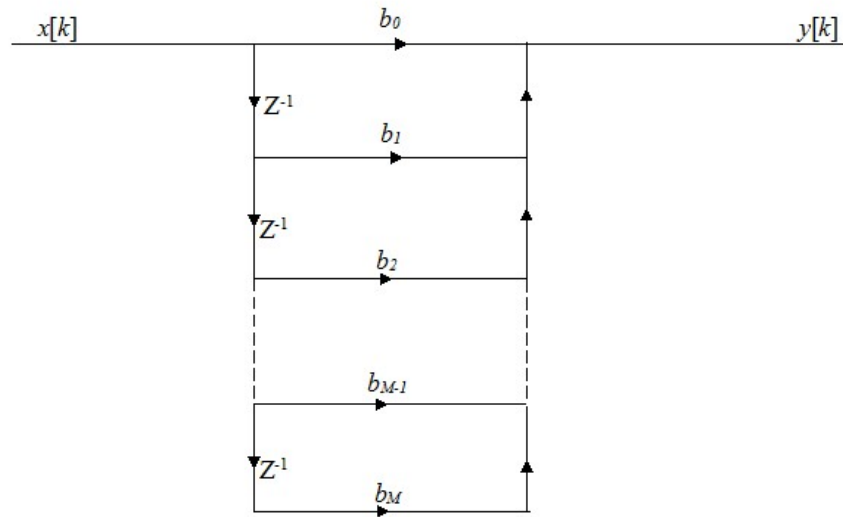
Figure 46. Signal flow graph for an Regular FIR filter system

Methods:

The key step of designing an FIR filter is the selection of order. After adjusted several parameters, we decided to apply a 400 order FIR notch filter (see Appendix 1.2) to the acquired data with the bandwidth of 0.1MHz since it gave out the best compromise between filtering effect and time consumption. If the order is too high, the loops will reduce the efficiency of the program. However, if the order is too low, the attenuation at the noise frequency will not be enough to satisfy the experiment requirement.

Results:

Figure 47 shows the RF signal with FUS noise processed by the FIR filter. The amount of noise didn't reduce as much as the IIR filter did.

Figure 48 indicates the frequency spectrum of the RF signal with FUS noise processed by the FIR filter. We can see that the value of the harmonic peaks decreased significantly (between 4-10MHz). So the noise shown in Figure 48 was mainly caused by the central frequency and low frequency harmonics of the FUS transducer.
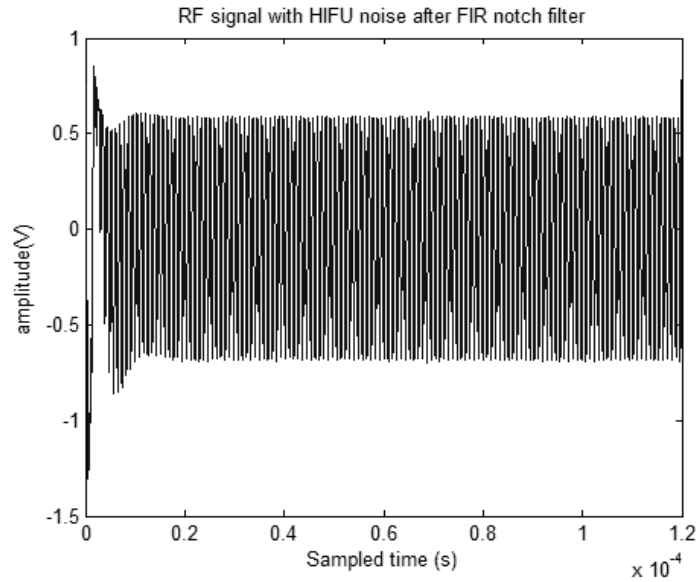
Figure 47. RF signal with FUS noise processed by the FIR filter
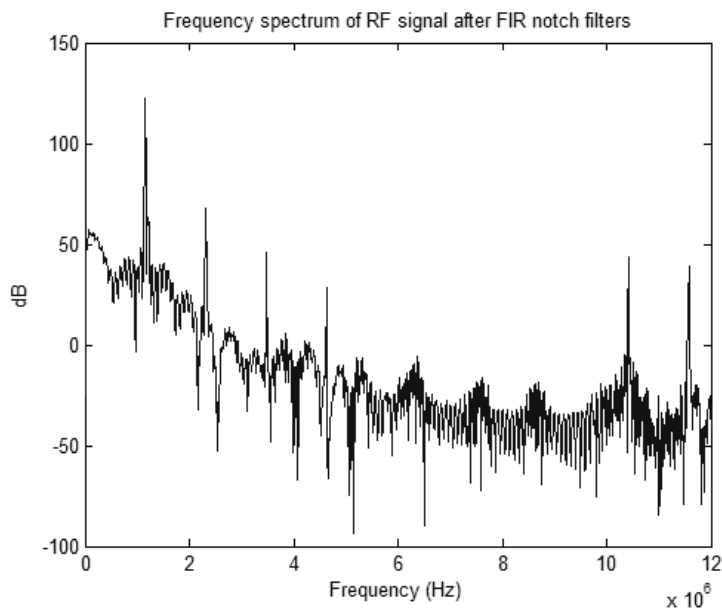
Figure 48. Frequency spectrum of RF signal with FUS noise processed by the FIR filter.

4.3.2 FIR Adaptive Filter

We modified the regular FIR filter to FIR adaptive Filter. An adaptive filter is a filter that self-adjusts its transfer function according to an optimization algorithm driven by an error signal [35]. Figure 49 shows the flow graph of adaptive filter, where:

$d[k]$: desired signal (reference);

$x[k]$: input signal;

$y[k]$: output signal;

44

$e[k]$: error signal;

W_k: weight (coefficients).
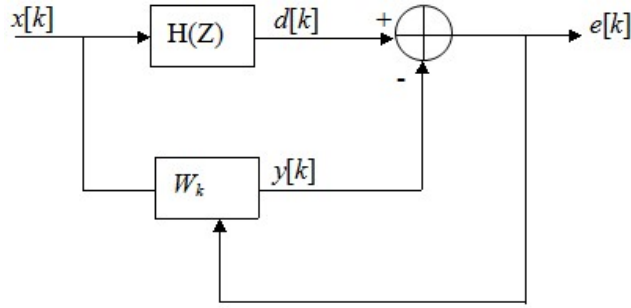
Figure 49. Signal flow graph for a Regular adaptive filter system

The error signal is the difference between the desired and output signal:

$$e[k] = d[k] - y[k] \qquad (4.3.2)$$

The output signal is estimated by the desired signal through convolving the input signal with the weight (coefficients).

$$y[k] = W_k * x[k] \qquad (4.3.3)$$

The adaptive filter updates the filter coefficients at every time using the error signal where ΔW_k is a correction factor for the filter coefficients.

$$W_{k+1} = W_k + \Delta W_k \qquad (4.3.4)$$

Methods:

In LHM experiment, we used the RF signal without FUS noise as reference ($d[k]$) which we acquired while the FUS transducer was off to filter the data with FUS noise. Details on the adaptive FIR filter can be seen in Appendix 1.3.

Results:

Figure 50 shows the RF signal with FUS noise processed by the adaptive filter. The signal looks almost the same with the one without FUS noise (Figure 23 (b)). Figure 51 indicates the frequency spectrum of the RF signal with FUS noise processed by the adaptive filter. In this case, we can see that all the harmonic noise peaks disappeared completely and the spectrum looks almost the same as the spectrum with the transducer off (Figure 43 (b)). Figure 52 shows the error signal.
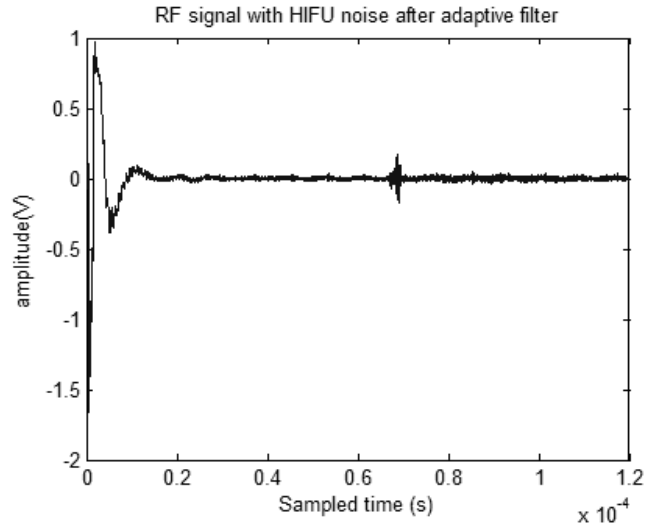
Figure 50. RF signal with FUS noise processed by the adaptive filter
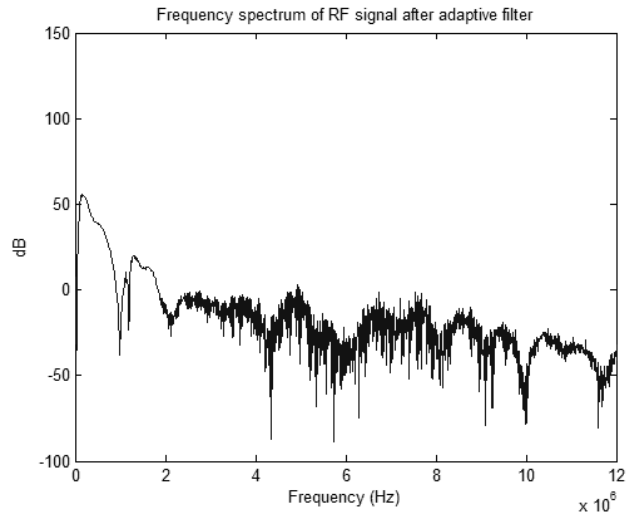
Figure 51. Frequency spectrum of RF signal with FUS noise processed by the adaptive filter.
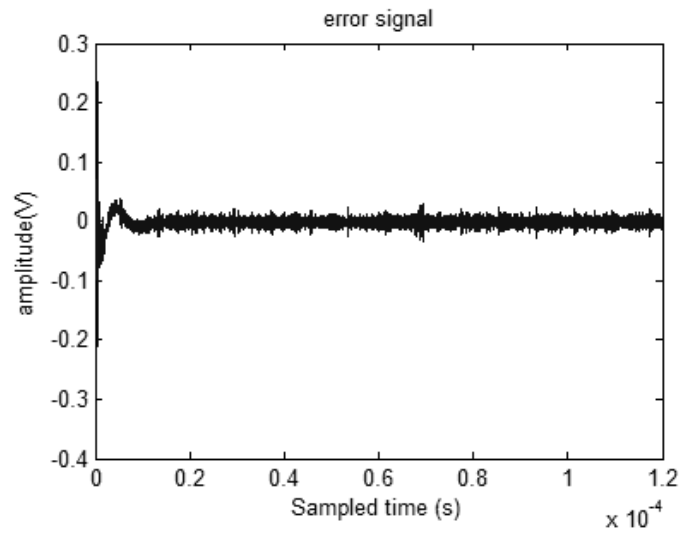
Figure 52. Error signal (difference from the desired signal and output signal)

4.4 Discussion:

4.4.1 Comparison of the three filters

The results of three different filters (IIR notch filter, regular FIR notch filter and FIR adaptive filter) indicate that due to the filtering effect, FIR adaptive filter displayed the best performance. However, the idea of the adaptive filter is to make the output signal as close as possible to the desired signal (with FUS transducer off) which may lose our localized harmonic motion information during processing since LHM is derived from comparing the difference from two adjacent RF signals (the details will be discussed in Chapter 5). So this filter is not feasible for our LHM experiment.

The comparison of the filtering results from IIR notch filter and regular FIR filter illustrates that even though the filter order FIR filter (400) was much higher than the one of IIR filter (2), the IIR filter still presented a more satisfying performance than the FIR filter because of the FIR filter's disadvantages:

1. An FIR filter's impulse response duration, although finite, may have to be very long to obtain sharp cutoff characteristics;

2. The design of FIR filters to meet specific performance objectives is generally more difficult than the design of IIR filters for a similar application.

   However, the FIR filter had some other significant advantages as follows:

1. FIR filters can easily be designed to have constant phase delay;

2. FIR filters implemented with non-recursive techniques will always be stable and free from the limit-cycle oscillations [34].

5. Measurement of LHM displacements

5.1 Introduction

After finishing the preparation work (experiment setup and filters in Chapter 2, 3 and 4), we started LHM displacement experiments. This chapter will cover the measurement of LHM displacements which includes the analysis of the noise contained in the displacement waveforms, the influence of analog and digital filters on the displacement detection and of other characteristics of the obtained displacements.

5.2 Tissue Motion Calculations

In order to calculate the displacement of the sample during the LHM induction, we acquire a series of RF signals during the motion (in our case 462 RF signals).

When tissues are static, the same RF signal is expected every time that we send and receive echoes, when they are moving we expect to receive the echoes before or after, depending if they are moving towards or away from the transducer.

We can therefore calculate the displacement by cross correlating RF signals acquired during motion and detecting the differences in time for the echoes, and converting them to distance by taking into account the speed of sound.

In signal processing, cross correlation is a measure of similarity of two waveforms as a function of a time lag applied to one of them. So we looked for a time delay between two adjacent (in time) RF signals.

Cross correlation can be calculated using Eq. (5.2.1), by shifting τ, the similarity was measure between signal x and y.

$$\rho(\tau) = \frac{C_{xy}(\tau)}{\sqrt{C_{xx}(\tau) \bullet C_{yy}(\tau)}}$$
(5.2.1)

Where

$$C_{xy}(\tau) = \sum_{-\infty}^{+\infty} x(t) \bullet y(t + \tau)\Delta t$$
(5.2.2)

$x(t)$ is signal1 and $y(t)$ is signal 2, τ is the time delay

So in our implementation, we first divide the RF signal into windows around the depth that we are exploring (close to the focus is where the displacements are expected). We then took a segment of the RF signal and calculated the normalized correlation $\rho(\tau)$ with the

previous RF signal for the same time segment. By finding the t delay that corresponds to the maximum in correlation, we obtained the displacement supposing that speed of sound is c: $d=ct/2$ (Figure 53).
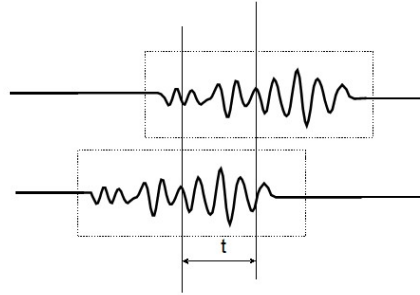
Figure 53. Delay t corresponds to the maximum in correlation which could be used in the calculation of displacement

Figure 54 shows an example of the LHM estimation using RF echo lines. The window (dashed box) was cross-correlated with the previous RF signal and displacement was obtained where the highest correlation was registered [27]. The frequency to send RF signal at t_1, t_2, etc. was 5 kHz which was referred as displacement calculation frequency.
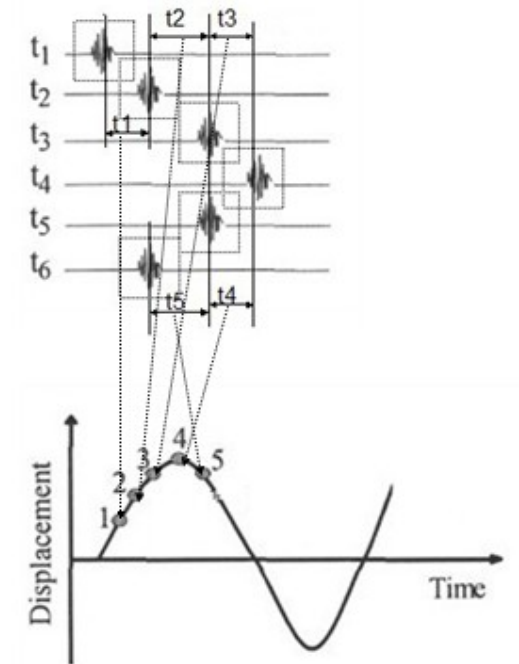
Figure 54. Concept of Harmonic Motion Imaging for displacement estimation. RF line tracking at different instants (t_1, t_2, etc.) acquired at the focus of the imaging transducer yields precise displacement estimated and identifies the characteristics of the locally induced vibration.

49

5.3 Displacement Analysis

During the first acquisitions and calculations of displacements, it was not possible to detect properly the value of displacements expected. In order to analyze and obtain results, we adjusted a series of parameters including peak to peak amplitude of the modulated signal, as well as the range of the acquisition card (ATS460).

Then we chose the setting for LHM experiment as 0.35V for peak to peak amplitude of the modulated signal and 4V for the range of ATS460. All the plots in this section were achieved using the low-pass filter, notch filters at 5th (5.79MHz), 7th (8.1MHz), 8th (9.26MHz), and 9th (10.42MHz) harmonics and also using a small signal amplifier and IIR Butterworth digital notch filter.

Method and Materials:

Analysis of the LHM at different depth:

Figure 55 shows the simplified setup schematic of LHM experiment. The focal length for both transducers was 60mm and the focus of the transducers was located inside the sample at 1cm depth. In order to get the best displacement results, we analyzed the acquired data at different depth (from 50mm to 69mm distance away from both of the transducers with the step of 1mm) which means we calculated the displacements at different depths from 50 to 69mm and obtained a 2D map of displacements.
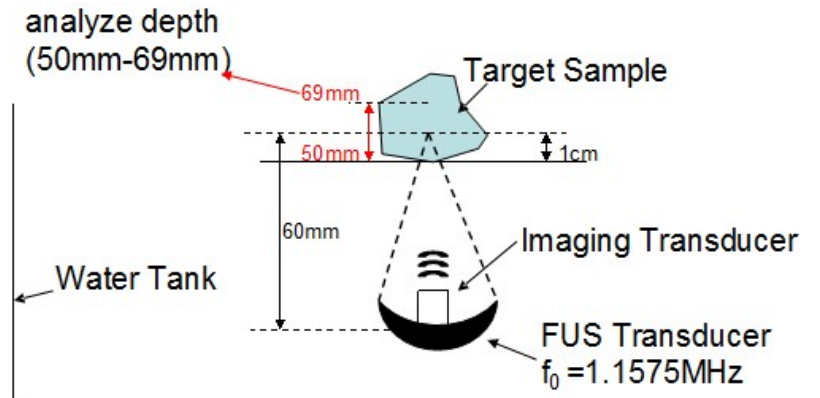
Figure 55. Simplified setup schematic of LHM experiment

Figure 56 shows the 2D image of 462 RF signals with the FUS transducer off (a) and on (b). In both conditions, we can see a line around 50mm which was the echo coming back from the target sample interface. In

Figure 56(a), when the FUS transducer was off, the RF signals we acquired were almost flat everywhere which means the sample was not in motion. Since the FUS transducer was off in this case, there was no localized harmonic motion inside the sample as expected.

It was expected to obtain an image very similar to the conditions without motion (

Figure 56(a)). However, the RF signal displays large differences that follow the noise. The clean signal should look similar to the RF signal obtained with the FUS transducer off since the displacements are small (usually around a maximum of 200μm) and that is not visible on the RF signal map without processing. This noise affecting the acquired RF signal will make it more difficult for us to accurately calculate the LHM displacements, as it will be discussed below. The explanation for the noise will be discussed in the discussion part.
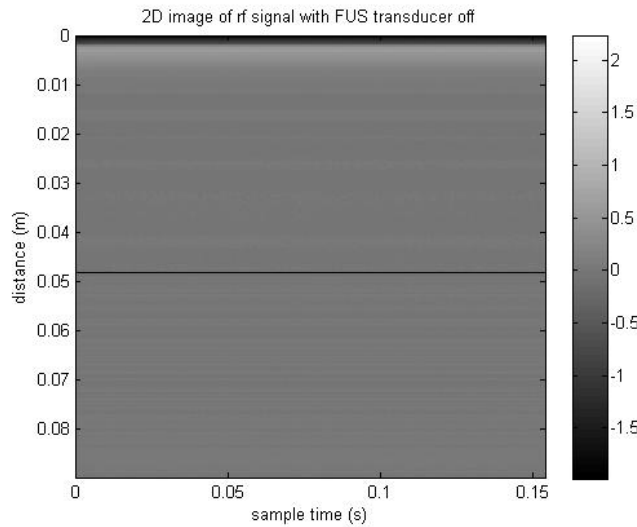
(a)
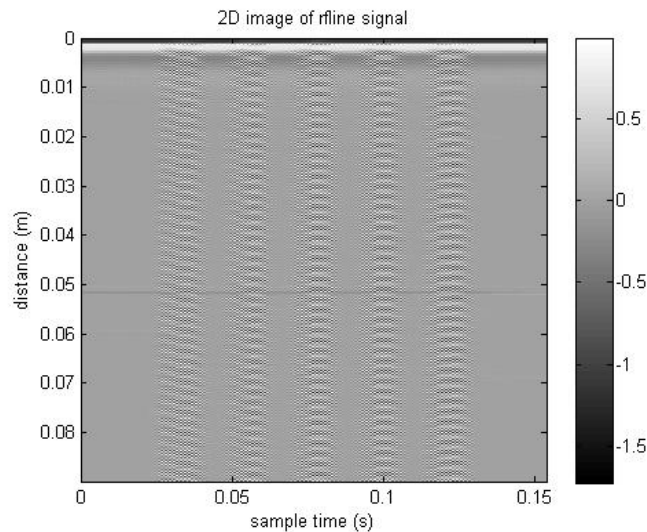
(b)

Figure 56. 2D image of 462 RF signals when target sample was placed at 50mm distance from the FUS transducer. (a) is acquired when the FUS transducer was off as reference while (b) was acquired when the FUS transducer was on.

51

In order to conduct a further analysis of LHM displacements, we conducted three analyses:

1)   1D image of single displacements;

2)   Spectrum of displacement;

3)   2D image of displacements.

Results:

(1) 1D image of single LHM displacement

To analyze a 1D image we only focused on the displacements at the focus. Figure 57 shows the 1D LHM displacements around the focus part (at 59mm depth) as a function of time. According to Figure 22, the first and last cycles of the signal we sent from the FUS transducer to the target sample were just flat lines which mean the transducer was not generating a force to push the sample, so we should not register any displacements within these two cycles. The LHM displacement plot below (Figure 57) shows first and last cycles of flat lines as we expected. We sent 5 cycles of amplitude modulated signals to induce localized harmonic motion inside the sample, and we can see 5 cycles of displacement in Figure 57 which proves that the 5 cycles of modulated excitation produced 5 cycles of displacement and they could be detected.
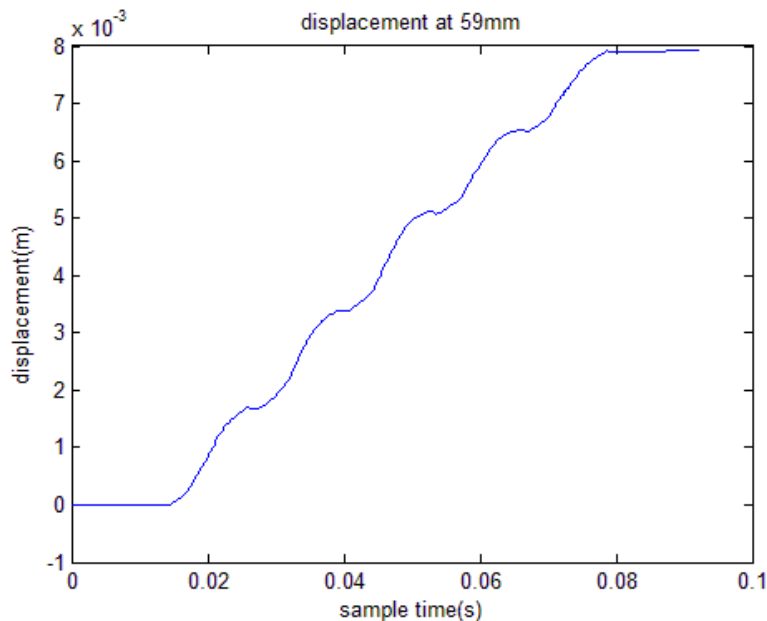
Figure 57. Single 1D displacement at 59mm depth

(2) Frequency Spectrum of Displacements

Through the observation of single 1D LHM displacement at 59mm distance (Figure 57), we can see that the displacement has a trend to go up so that we can predict there is a large

low frequency component in the frequency spectrum of the displacements.

Figure 58 shows the frequency spectrum of displacements at all analyzed depths (from 50mm to 69mm). As we can see, the spectrums at different analyzed depths all have a large low frequency component. It made the observation of displacement amplitudes (usually around 200μm as talked before) calculated from localized harmonic motion very difficult (shown in circle below).
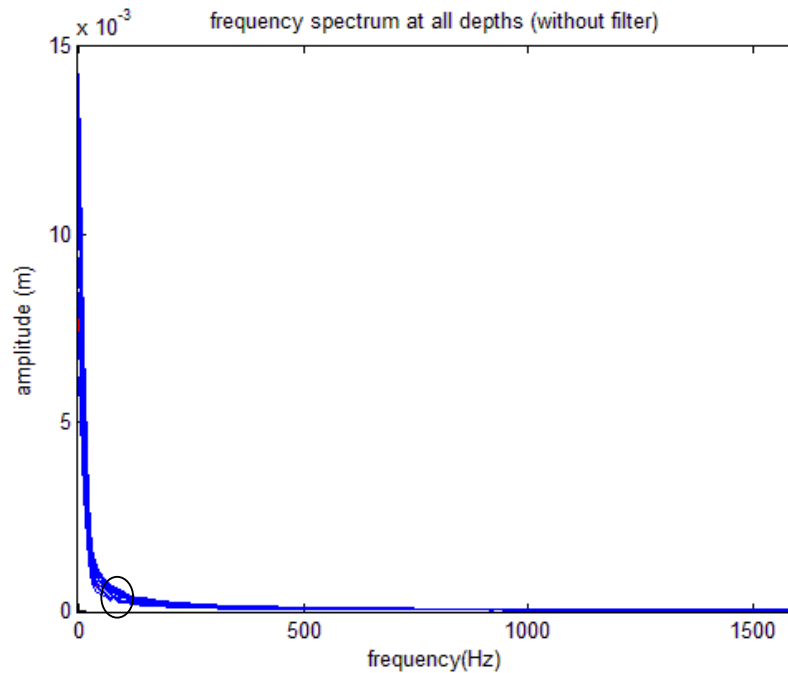
Figure 58. The frequency spectrums of displacements at all depths (without digital filters)

So in order to achieve a cleaner LHM displacement plot, we decided to add a software low-pass and a software high-pass filter on the displacements using MATLAB. Since the sample should carry out the LHM at 75 Hz following the modulation frequency (75Hz), our attention is only on the frequency around 75Hz. So In order to cut off the noise far away from 75Hz, we implemented a low-pass and a high-pass 5th order Butterworth filter with the cut off frequency at 225Hz and 63.75Hz respectively. Figure 60 shows the result of frequency spectrums of displacements at all depths after the Matlab filters. As our focused ultrasound signal was modulated at 75Hz, we expect the target sample to undergo a LHM at 75Hz. So we should focus on the amplitude of the frequency around 75Hz. Since the displacement calculation frequency is just 5 kHz, the frequency spectrum for the displacements has a low resolution (5 kHz) which could only allow us to get the amplitude at 78.13Hz instead of the exact amplitude value of 75Hz.
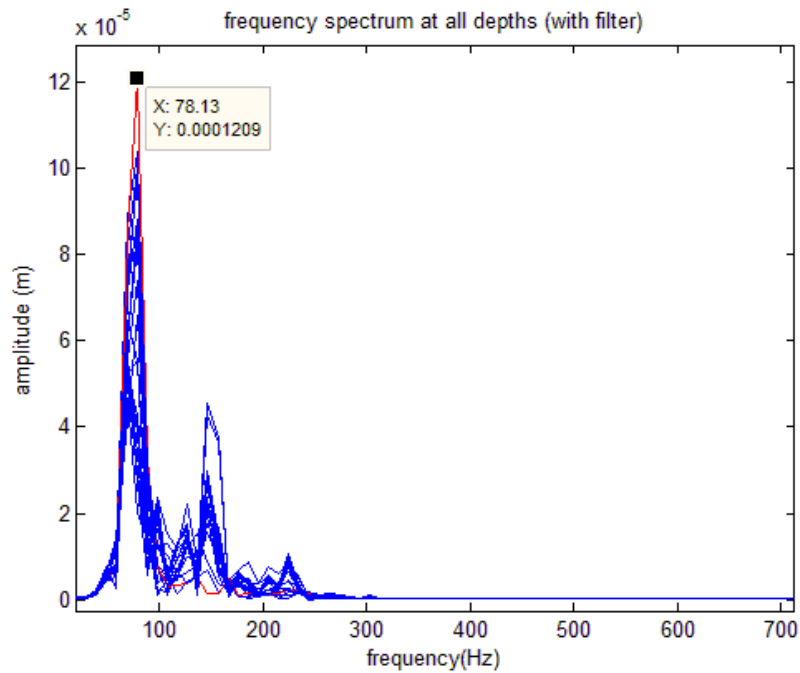
Figure 59. The frequency spectrums of displacements at all depths (with matlab filters)

As shown in Figure 59, since the Matlab filters already filtered the high and low frequency noise, the peak of the amplitude appeared at 78.13Hz as expected which proved that the movement we tracked was the localized harmonic motion induced by the modulated excitation. The frequency of this harmonic motion was around 75Hz and the maximum amplitude shown with tag beside was 120.9μm (achieved at 59mm distance from the transducers) in this case.

(3) 2D image of displacement

When the distance from FUS transducer and target sample is 50mm, Figure 60 and Figure 61 below illustrate the 2D image of displacements as a function of time (horizontal axis) and at different depths (vertical axis).

For the same reason outlined above, the large low frequency component noise led to large displacement values and we were not able to clearly observe 5 cycles of motion in the 2D image of displacement (Figure 60). So we still need to implement the same Matlab software filters to analyze the 2D image of displacement. In order to increase the signal to noise, we averaged the displacements obtained over 5 different experiments.
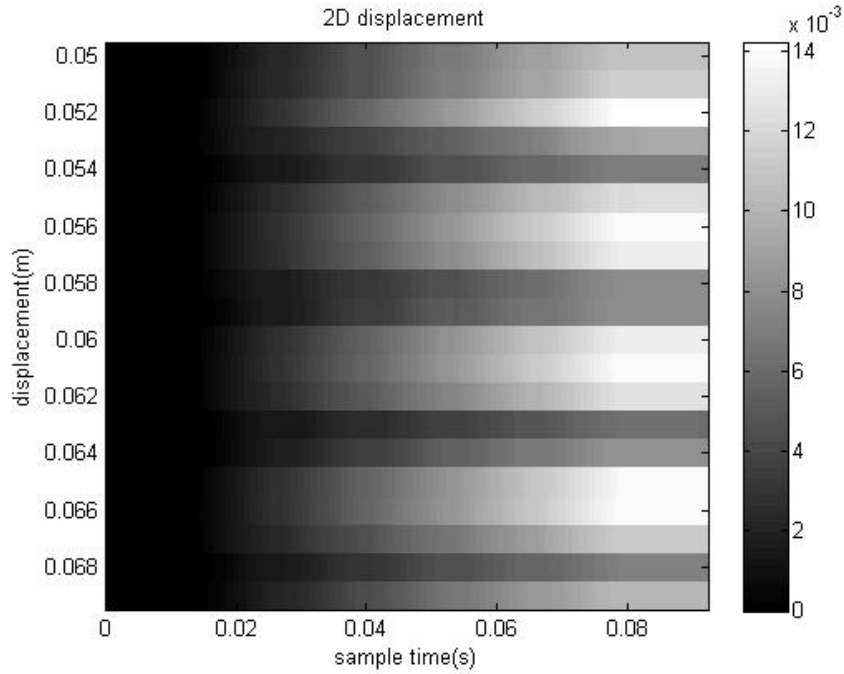
Figure 60. The 2D image of displacements of different depths when the distance from sample and FUS transducer was 50mm without Matlab filter

After we added both of the Matlab software filters to 2D LHM displacement image analysis, the displacements were observed more accurately. Each displacement has 5 cycles of bursts as expected. So it can also prove that 5 cycles of modulated wave were sent to the target sample and 5 displacements have been detected (Figure 61).

Moreover, as we can see, not all the depths showed a very clear displacement. The main reason was because they had a too high low frequency component in those lines that overshadowed the 75Hz so it was not possible to obtain the actual displacement (details will be discussed in 5.3.2.2).
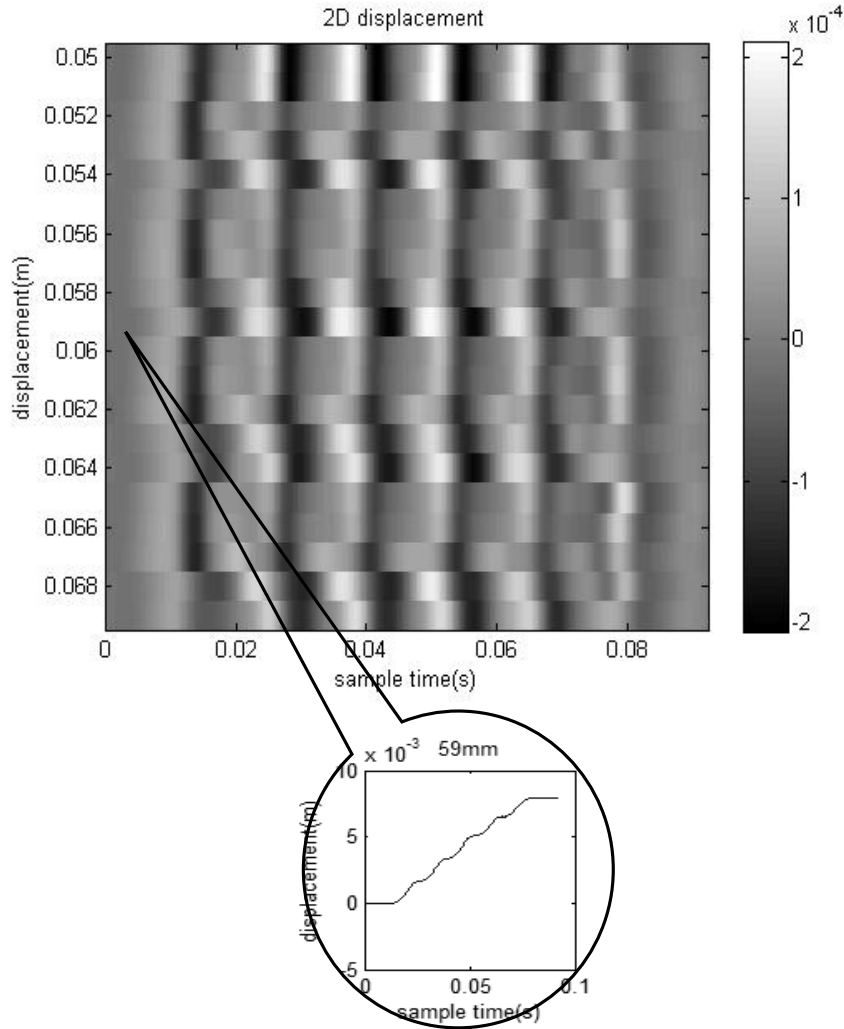
Figure 61. The 2D image of displacements of different depths when the distance from sample and FUS transducer was 50mm with Matlab filters after 5 times of averaging

5.3.1 Displacement Testing Results with Different Analog Filters.

As discussed in Chapter 3, we implemented analog filters (including low-pass, high-pass and notch filters) and we tested their impact of using the filters to the displacement measurements. We applied different filter combinations to the LHM experiment to obtain the testing results. In this way, we could have an overall look at the filter impact with different filters combination in order to select best setup.

Method and Materials:

The combinations included following cases:

Low pass filter and notch filter at 5.79MHz (LPN5);

Low pass filter and notch filter at 5.79 and 6.95MHz (LPN5N6);

Low pass filter and notch filter at 5.79 and 8.10MHz (LPN5N8);

Low pass filter and notch filter at 5.79 and 9.26MHz (LPN5N9);

Low pass filter and notch filter at 5.79 and 10.42MHz (LPN5N10);

Low pass filter and notch filter at 5.79, 6.95 and 8.10MHz (LPN5N6N8);

Low pass filter and notch filter at 5.79, 6.95 and 9.26MHz (LPN5N6N9);

Low pass filter and notch filter at 5.79, 6.95 and 10.42MHz (LPN5N6N10);

Low pass filter and notch filter at 5.79, 8.10, 9.26 and 10.42MHz (LPN5N8N9N10);

Low pass filter and notch filter at 5.79, 6.95, 8.10, 9.26 and 10.42MHz (LP_ALL);

We did 5 tests for each case and counted the number of times that a successful displacement was obtained (Table 9 (a) and (b)). The criteria we used to judge whether a displacement was successful or failed was its frequency spectrum. If a displacement was successfully measured, the spectrum should show a peak around 75Hz which was the LHM frequency. However, if it failed, we could hardly detect the peak since the low frequency component noise was too strong and covered the LHM frequency.

If we zoom in the frequency spectrum of displacements without filter (

Figure 62) under the condition of LPN5N8N9N10, we can see the solid line (frequency spectrum at a depth of 59mm) which has an obvious peak around 75Hz while the dashed line doesn't (frequency spectrum at a depth of 57mm). Correspondingly, in the plot processed after matlab filters (Figure 63), the peak amplitude of frequency spectrum at 57mm depth was not at the expected frequency and gave out a much smaller value (38.87µm).
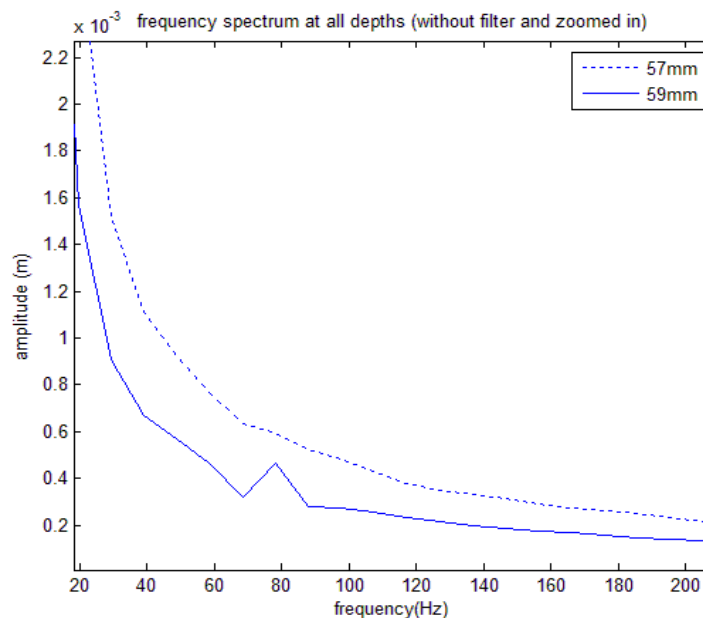
Figure 62. Frequency spectrum at 57mm and 59mm (without matlab filters and zoomed in)
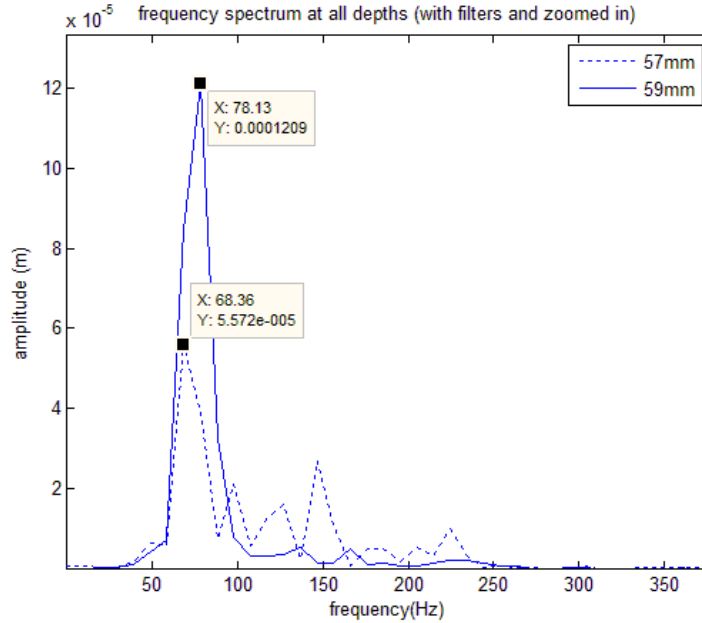
Figure 63. Frequency spectrum at 57mm and 59mm (with matlab filters and zoomed in)

We concluded that under the LPN5N7N8N9 condition, the displacement calculation at 59mm depth was successful while the one at 57mm depth failed.

Table 9 (a) was the testing result when we used the small signal amplifier (AU 1583, 35dB gain, MITEQ, NY, USA) while Table 9 (b) was acquired when we used the amplifier of the pulser/receiver.

The small signal amplifier has the gain of 35dB and Table 8 shows the parameter selection of pulser/receiver (the gain of amplifier inside the pulser/receiver, pulse energy, damping).

Gain (PR):	45dB
Pulse Energy (PR):	Low 4
Damping (PR):	15

Table 8. Setup for pulser/receiver in LHM experiment

We analyzed the results at 57, 58 and 59 mm from the transducer surface, with the geometrical focus at 60 mm (10 mm inside the sample).

Results:

Table 9 (a) and (b) shows successful ratio out of the 5 times of testing results for each case and also under the two conditions: small signal amplifier and the amplifier inside the pulser/receiver. Successful displacement calculations according to the criteria described

above, over total calculations were counted.

Testing Result with Small Signal Amplifier			
Depth Case	57mm	58mm	59mm
LPN5	5/5	5/5	0/5
LPN5N6	5/5	5/5	0/5
LPN5N8	5/5	5/5	0/5
LPN5N9	5/5	5/5	0/5
LPN5N10	5/5	5/5	0/5
LPN5N6N8	3/5	5/5	0/5
LPN5N6N9	3/5	5/5	0/5
LPN5N6N10	2/5	5/5	0/5
LPN5N8N9N10	0/5	5/5	5/5
LP_ALL	0/5	0/5	3/5

(a)

Testing Result with Pulser/Receiver			
Depth Case	57mm	58mm	59mm
LPN5	0/5	5/5	0/5
LPN5N6	0/5	5/5	5/5
LPN5N8	0/5	5/5	5/5
LPN5N9	0/5	5/5	4/5
LPN5N10	0/5	5/5	5/5
LPN5N6N8	0/5	5/5	5/5
LPN5N6N9	0/5	5/5	5/5
LPN5N6N10	0/5	5/5	5/5
LPN5N8N9N10	0/5	4/5	4/5
LP_ALL	0/5	3/5	4/5

(b)

Table 9. Testing result for different combinations of filters.

After comparing these two groups of data, we notice that for the data obtained using the small signal amplifier (Table 9 (a)), at 57mm and 58mm in depth, we could calculate the

displacement successfully until LHN5N8N9N10. After this case, the successfully calculated displacements shifted from 57 and 58mm to 58 and 59 mm. Maybe because the sample moved since these experiments were performed at different times.

The success ratio for displacements under these two conditions (small signal amplifier and pulser/receiver) looks similar. However, considering the fact that the first group data was acquired with a 35dB gain while the gain for second group of data was 45dB, we can conclude that the small signal amplifier gave a better performance than the amplifier inside the pulser/receiver. So we decided to choose small signal amplifier for the following LHM experiments.

According to Table 9 (a), at 57mm in depth, the success ratio for LPN5N6N8, LPN5N6N9 and LPN5N6N10 was 3/5, 3/5 and 2/5 respectively. This success was lower than when using LPN5N8, LPN5N9 and LPN5N10 whose ratio was all 5/5. The only difference between these cases was the notch filter at 6.95MHz. And also, if we compare the results from LPN5N8N9N10 and LP_ALL, the notch filter at 6.95MHz was also present and the performance decreased. So this filter didn't seem to improve the data, instead, it had a bad effect. This result could be explained as outlined in the discussion in Chapter 3. Due to the large bandwidth of the notch filter at 6.95MHz; it deformed the signal too much which led to a deformed RF signal and a lower performance in cross-correlation and therefore displacement calculation.

So to conclude, for the following LHM experiment, we chose the set up using the small signal amplifier with low pass filter and notch filters at 5.79, 8.10, 9.26 and 10.42MHz.

5.3.2 Discussion

5.3.2.1 Analog High Pass Filter

Even if the testing result in chapter 3.2.5 shows a good performance of the high pass filter and it got rid of most of the low frequency noise, in the LHM displacement experiment and calculation, we didn't use it because it had an impact on the signal phase which is the key parameter cross-correlation technique which leads to the failure of displacement detection. So we decided to remove it from the LHM experiment set up.

This led to a bad effect that the low frequency harmonics noises (especially the central frequency) still remained in the RF signals which increased the difficulty to calculate LHM displacement.

5.3.2.2 Displacement at different depths

Theoretically, we should detect LHM displacements around the geometrical focus (from 57mm to 63mm). At the other depths which are farther away from the FUS transducer focus, the force is reduced and the amplitude of the LHM displacements should be gradually smaller. However, we could only detect displacement at several depths intermittently.

Figure 64 shows some of the single 1D displacement plots at different depths (51, 54, 58, 59, 60, 63, 66, 68 and 69mm) and among these plots, we successfully detected the displacements at 54, 58, 59, 63, 68, and 69mm while we could not obtain obvious displacement from the others. This result could also be observed on the 2D image of displacement (Figure 61). The one with a more obvious displacement shows a better contrast of the 5 bursts.

(a)

(b)

(c)

(d)
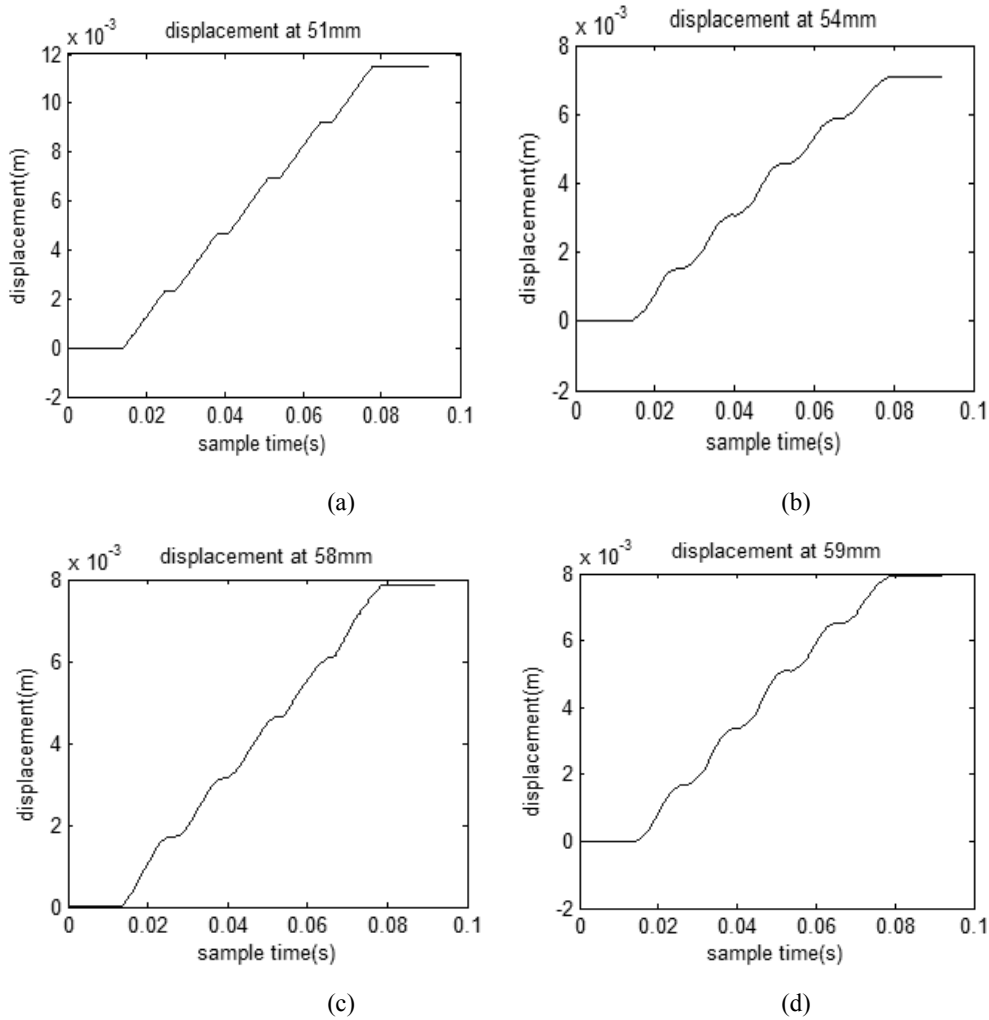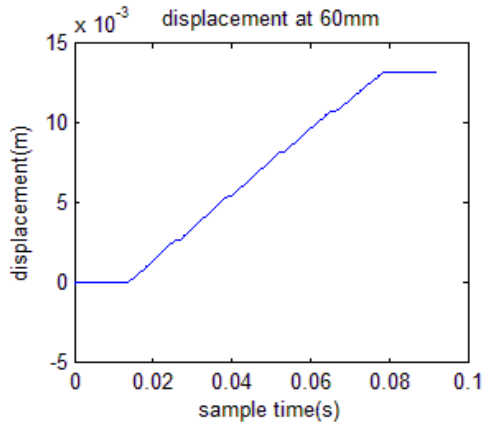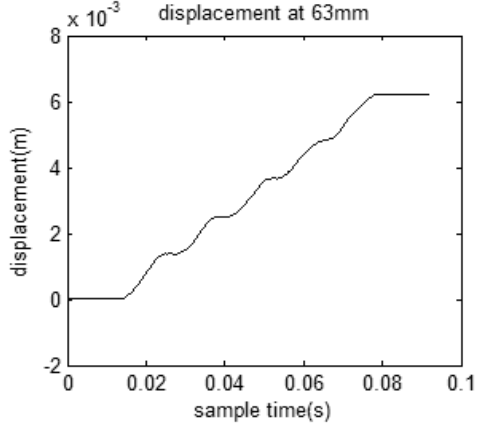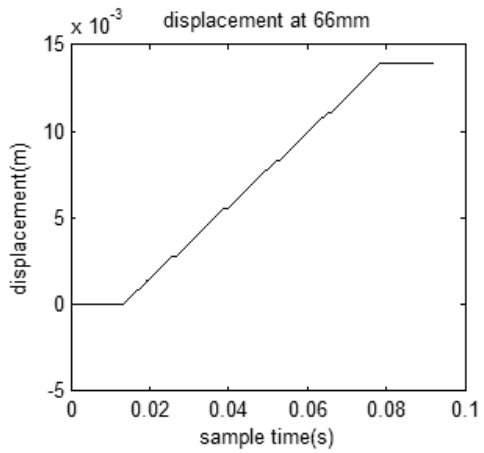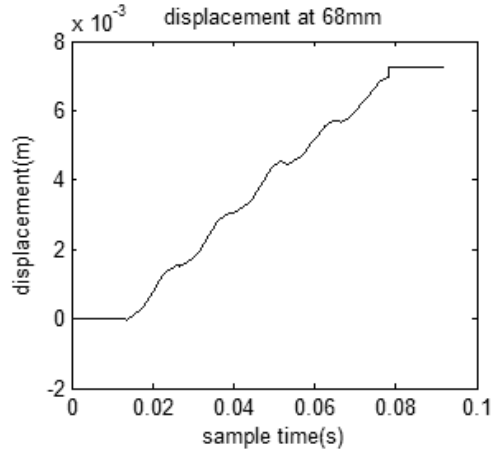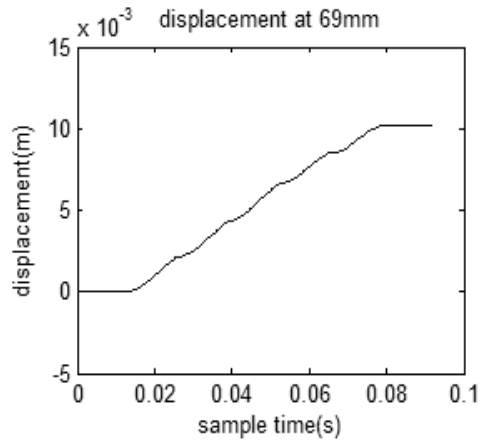
Figure 64. Single 1D displacement at several different depths (51, 54, 58, 59, 60, 63, 66, 68 and 69mm)

For some of the depths where we could not register displacement, it may be because it was too far from the focus (like 51mm) and the motion at that depth was not strong enough to

be detected. However, since we could detect LHM displacement at 69mm, we expected to register displacement at 51mm due to the symmetrical distribution of the energy field. Furthermore, for the depths which were just at the focus (60mm), we should definitely detect displacement and it was not possible.

There were two explanations for the failure of displacement detection: electrical noise or mechanical/acoustic noise.

(1). Electrical noise.

Electrical noise refers to the noise brought by the system itself such as cables or equipment. Figure 65 and Figure 66 show the spectrum and 2D image of a single RF signal respectively acquired with no FUS transducer (a 50Ω load was used instead). Both figures show a very clean signal which means there was no pure electrical noise. So we concluded that most of the registered noise was caused by mechanical or acoustic sources.
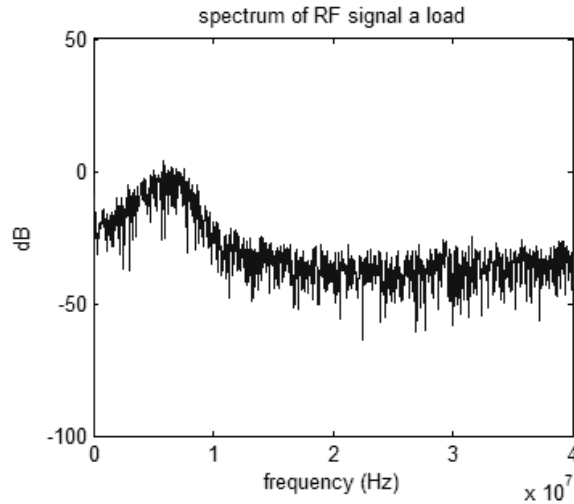
Figure 65. Spectrum of single RF signal with a 50Ω load
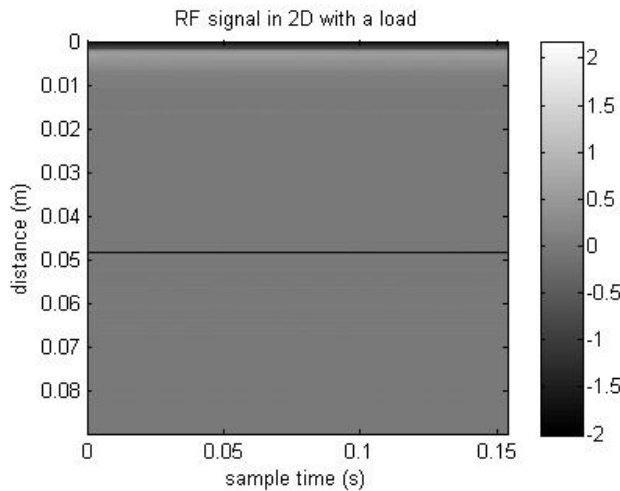
Figure 66. 2D image of single RF signal with a 50Ω load

(2). Mechanical/Acoustic Noise.

According to

Figure 56 (b), we already noticed that the RF signals contained a large amount of noise which could lead to failed displacement detection. Mechanical/Acoustic noise is the noise brought by FUS transducer. It was caused either by the reflection of the ultrasound wave at interfaces (like water or plastic holder) or by the vibration of the imaging transducer since it is mechanically coupled to the FUS transducer. The imaging transducer captured this and rendered them as electrical noise on the RF signal. Figure 67 shows the frequency spectrum of RF signals with FUS transducer on. Since we decided to remove both the high pass filter and notch filter at 6th harmonic, we can see that the noise peak at 1st, 2nd, 3rd, 4th and 6th harmonics displayed a higher value than other harmonics since no filters took care of them.
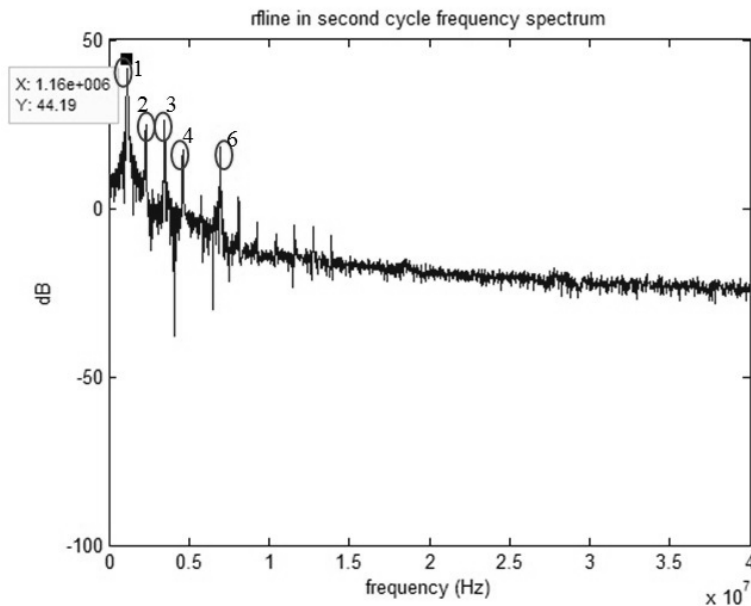
Figure 67. Frequency spectrum of RF signal in second cycle

5.3.3 Displacement Testing Result with Different Digital Filters

5.3.3.1 Adaptive Filter

As discussed before (Chapter 4.3.2), the idea of adaptive filter is to make the output signal as close as possible to the desired signal. Since we used the RF signal with FUS transducer off as reference in this case, all the RF signals will be displayed similarly to the RF signals which did not contain the localized harmonic motion information.

The algorithm of cross-correlation could not detect the changes between present RF

signal and the previous RF signal. Therefore we lost all the displacement information during the filtering process. Even though it gave the best filtering effect among the three filter types (Figure 51), it is infeasible in the LHM experiment.

5.4.3.2 IIR Notch Filter

All the displacements shown above (Figure 64) were acquired using IIR notch filter, but not all the depths showed a very clear displacement. The main reason was that we decided to remove the analog high pass filter; the central frequency and other low frequency noises were still left in the acquired RF signals. So the displacements had a too high low frequency component in those lines that overshadowed the 75Hz so it was not possible to obtain the actual displacement.

Due to the fact that the attenuation of IIR filter at the central frequency and first few harmonic frequencies was not big enough to get rid of all the noise, the displacement result did not behave very stable.

5.4.3.3 FIR Notch Filter

The performance of regular FIR was even worse than the IIR filter for the central frequency and the first few harmonic noises.

Figure 68 and Figure 69 show the displacement at 59mm in depth without any digital filters and with FIR notch filter. We couldn't register any displacement without filters, however, after processed by the FIR notch filter, the shape of the 3rd cycle of the displacement showed up. However, we were not able to achieve a complete displacement plot under this condition.

The testing result showed that without the analog high pass filter, this FIR notch filter could only improve the displacement file but could not give us a clear displacement plot. This result also verified the disadvantage of FIR filters: the design of FIR filters to meet specific performance objectives is generally more difficult than the design of IIR filters for similar application.
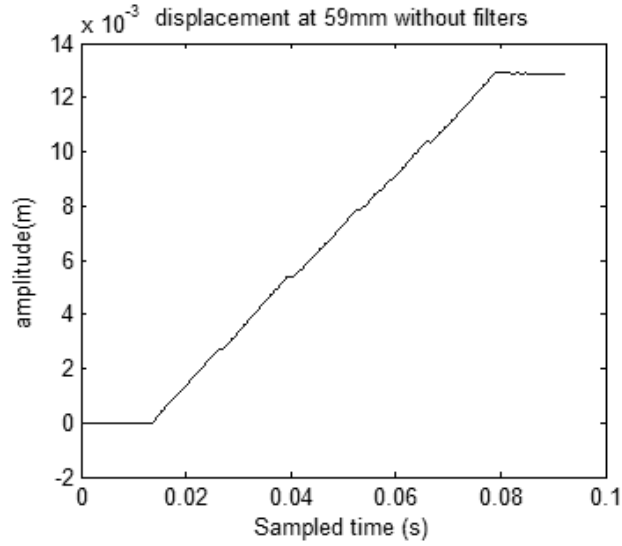
Figure 68. Displacement at 59mm analyze depth without digital filters
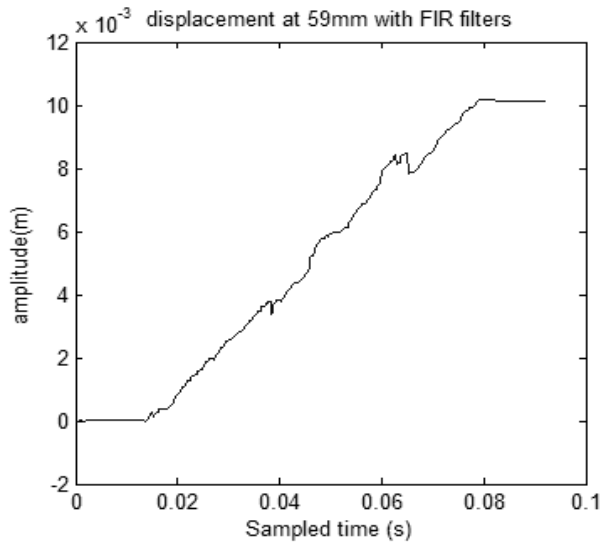
Figure 69. Displacement at 59mm analyze depth with FIR notch filter

5.4.4 Conclusion

Because of the lack of analog high pass filter, IIR notch filter is the best choice in this condition. And FIR filter is not suitable for strong noise situation since it could not achieve attenuation as high as the IIR filter.

However, in the small noise condition, FIR filter has its own advantages such as it is developed with non-recursive techniques which mean that the present state of the output does not relate to the previous state of the output. This could allow us to implement the FIR filter into parallel programming so that the speed of processing will increase significantly (details will be discussed in Chapter 7.3).

6. Measurement of LHM displacements in samples with different hardness

6.1 Introduction

In LHM experiment, we use the elasticity (Young modulus) of samples to estimate their acoustic characteristics. The sample with a lower Young modulus, which means it is harder, will undergo a smaller distortion under the same pressure and carry out a harmonic motion with smaller amplitude. This is the main idea of Localized Harmonic Motion Imaging to differentiate tissues with different Young modulus.

So this chapter will discuss LHM displacements in samples with different hardness and compare the amplitudes of LHM.

6.2 LHM displacement detection in samples with different hardness

Methods:

We used silicon as our phantoms to determine the hardness differences. Room-temperature-vulcanizing (RTV) silicon (RTV6166; two-part silicone, Momentive Performance Materials, New York, NY, USA) was used to make all the phantoms since the two-part silicone has a similar behavior to tissues. By varying the proportions of part A and B on the silicon mix before vulcanization it was possible to vary the obtained hardness. A higher proportion of part A to part B presents a lower Young modulus which means that the sample becomes harder [36] .

The silicon mix was poured into a cylindrical mold of 45mm diameter by 75mm in height and heated at 70ºC for 1 h to attain vulcanization. For scattering purposes, acid-washed glass beads (particle size <106μm, Sigma-Aldrich, Oakville, ON, Canada) at a concentration of 2.5% by weight were added to the silicon mix before vulcanization.

Five sets of phantoms with different part A to B silicon proportions were made and tested (50:50; 40:60; 30:70; 20:80; 10:90). For this silicon mix, a proportion of part A to part B of a 40:60 presents a Young modulus of 7.63kPa while a 30:70 mix is harder with a modulus of 15.3kPa [36]. LHM was induced and amplitudes were obtained in these 5 phantoms.

The focus of transducer was located at 1 cm deep inside all the samples like before. And the amplitudes were calculated from using Fast Fourier Transform (FFT) to displacements data. The amplitude of the motion was obtained from the modulus of the FFT coefficient corresponding to the modulation frequency of 75 Hz.

Results:

Figure 70 shows the frequency spectrum of LHM displacement induced on the 5 sets of phantoms. The LHM displacement amplitudes were acquired from the displacement data around the focus (60mm) and also after averaging five measurements which could help reducing noise on the final calculated amplitude.
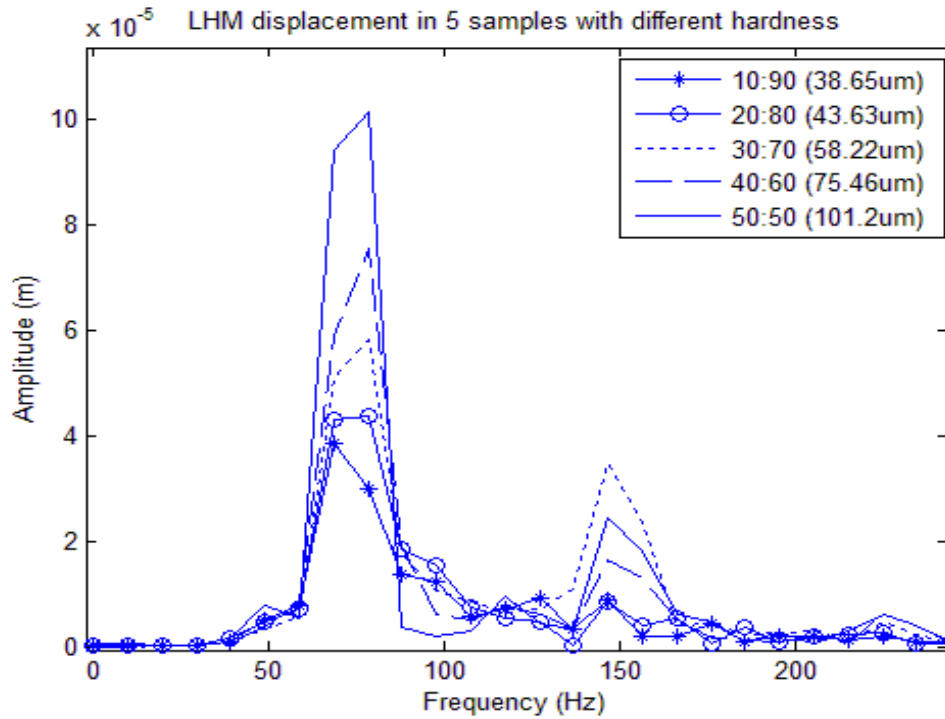
Figure 70. Frequency spectrum of Localized harmonic motion (LHM) displacements induced on the 5 sets of phantoms with the modulation frequency of 75Hz.

As it shows in Figure 70, the highest peak of LHM displacement amplitudes were around 75Hz (modulation frequency) as expected. The 5 sets of phantoms with different part A to B silicon proportions carried out LHM displacements of different amplitude which is shown in Table 10. Since a higher proportion of part A to part B presents a lower Young modulus, it displays a lower value of LHM displacement amplitude. According to this method, we can differentiate the parts with different stiffness.

Part A :Part B	Amplitude (μm)
10:90	30.65μm
20:80	43.63μm
30:70	58.22μm
40:60	75.46μm
50:50	101.2μm

Table 10. LHM displacement amplitudes in different phantoms.

6.3 Discussion

As we discussed in Chapter 5.3, since the displacement calculation frequency is just 5 kHz, the frequency spectrum for the displacements has a low resolution (5 kHz) which could only allow us to get the amplitude at 78.13Hz instead of the exact amplitude value of 75Hz. The results show the value we expected and an amplitude that decreased as the stiffness of samples increased. All the displacements showed a big peak around 75Hz which is the

modulation frequency.

Higher frequency peaks in the spectrum can be observed in Figure 70 (150Hz). There is an explanation for this phenomenon. It may be caused by the sharp slope when the radiation force is applied as compared with the slower relaxation when the force is stopped, making the displacements curve approach to a saw-tooth shape therefore generating higher frequencies in the FFT [36].

Another phenomenon has been observed due to different stiffness of samples, the softer samples displayed lower heights since they spread more. However, we still had enough height of the samples to locate the transducer focus at 1cm depth.

7. Program Optimization

7.1 Motivation

Due to the large amount of RF signals (462), it will cost large amount of time during LHM displacement calculation. Furthermore, these 462 RF signals only form the displacement information for a single point. If we want to obtain a whole image, it will take much more time (around 1h). Among the algorithms, the IIR filter was the most time consuming part during processing. So in order to increase the efficiency of the process, we paralleled the program with graphics processing unit (GPU). This chapter covers the introduction to GPU, the process of program optimization and the speed up we achieved before and after the modification.

7.2 GPU Introduction

Graphics Processing Units (GPUs) have emerged as a powerful platform for high-performance computation. They have been successfully used to accelerate many scientific workloads. Typically, the computationally intensive parts of the application are offloaded to the GPU, which serves as the CPU's parallel coprocessor [37].

Therefore, GPU has evolved into a highly parallel, multithreaded, many-core processor with tremendous computational horsepower and very high memory bandwidth compared to CPU, as illustrated by Figure 71 [38].

In November 2006, NVIDIA introduced CUDA, a general purpose parallel computing architecture – with a new parallel programming model and instruction set architecture – that leverages the parallel compute engine in NVIDIA GPUs to solve many complex computational problems in a more efficient way than on a CPU. With CUDA, computations are evolving from "central processing" on the CPU to "co-processing" on both the CPU and GPU, thus dramatically increase the computing performance.

CUDA has several advantages over traditional general-purpose computation on GPUs (GPGPU) using graphics APIs:

✓ Scattered reads – code can read from arbitrary addresses in memory;
✓ Shared memory – CUDA exposes a fast shared memory region (up to 48KB per Multi-Processor) that can be shared amongst threads. This can be used as a user-managed cache, enabling higher bandwidth than is possible using texture lookups.
✓ Faster downloads and readbacks to and from the GPU;
✓ Full support for integer and bitwise operations, including integer texture lookups [38].
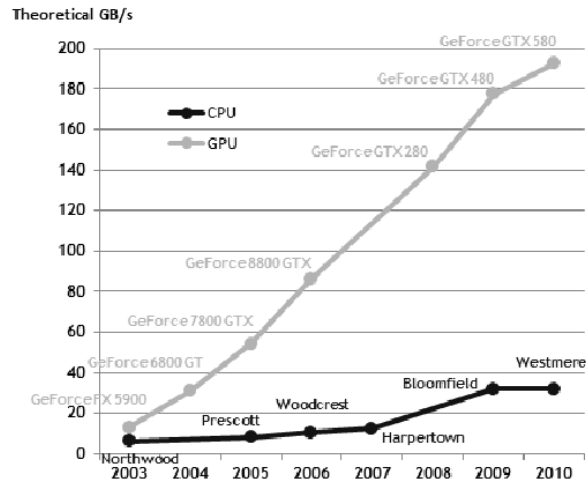
(a)
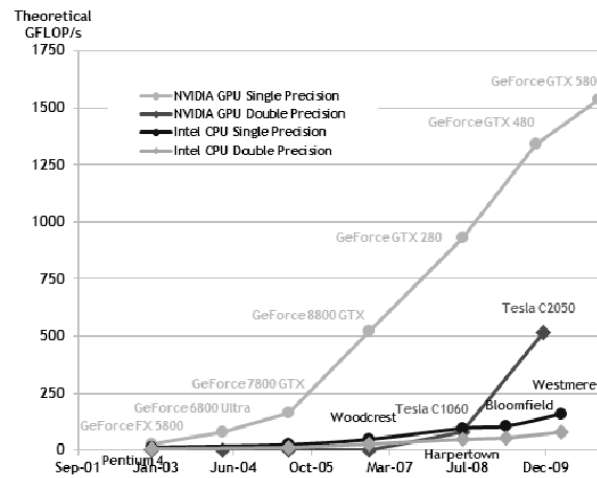
(b)

Figure 71. Floating-Point Operations per Second and Memory Bandwidth for the CPU and GPU [37]

CUDA has been widely used in those computation intensive applications including image and video processing, computational biology and chemistry, fluid dynamics simulation, CT image reconstruction, seismic analysis, ray tracing, and much more. Also, it is enthusiastically received in the many areas of scientific research, and many widely adopted commercial codes have been developed to use GPU Computing. There are many ways to make use of GPU Computing. In our project, we wrote and modified the code in C++.

7.3 Program Modification

A standard calculation needs the following steps:
1) Acquire all RF lines;
2) Filter all RF lines;
3) Cross-correlate the filtered RF lines to obtain displacements.

Since the part that consumes more time is the filtering, we concentrate on optimizing this algorithm.

Two optimizations were performed for this project:
1) Tried a FIR filter for a single RF line.
2) Then we tried to perform the filtering of all RF lines at the same time with IIR filter.

Methods:

FIR filter

As we know that FIR filter requires no feedback which makes the implementation simpler in GPU. In our experiment system, the order of FIR filter was 400 which is smaller than the maximum number of threads within one block (512). So the complexity of GPU programming will not be so difficult. The 400 order FIR filter allowed us to transfer the 400 coefficients to shared memory within one block.

During FIR filter CUDA implementation, we transferred all the coefficients into shared memory with one thread loading one of the coefficients. (For the full version of the code, please refer to Appendix 2.1))

```
__shared__ float s_coeficients[400];
if(tid<cn[i])
    s_coeficients[tid]=coeficients[coefDisp+tid];
```

The input signal ('rfLines[id]') were stored in global memory. It is because for each element of the output signal, it needed the contribution of 400 consecutive input elements. As we know that data in shared memory can only be accessed by its own block and they could not communicate with each other. So if we store the input signal in shared memory, the output signal may not be able to read the input data from other blocks. Also, even though the input signal could act as read only data, the length may vary from different situations, so we choose not to store it in constant memory. The algorithm of the FIR filter (Equation 4.3.1.1) was implemented by the code below:

```
if(id<cn[i]){
    stmp_rfLines[id]=32500;//normalized '0';
}
else if (id>=cn[i] && id< rfLen)
{
    for(int j=0; j<cn[i]; j++){
        stmp_rfLines[tid]+=s_coeficients[coefDisp+j]*rfLines[id-j];
    }
}
```

In our LHM experiment system, the length of the RF signals we wanted to filter was 3433. We assigned 512 threads to each block, so that 7 blocks were needed in total. Each of the threads was in charge of calculating one element of the RF signal. The final results were transferred back to the global memory as shown below (stmp_rfLines[tid] was a temporary variable declared in shared memory for saving the temporary value of the output signal).

```
rfLines[id]=stmp_rfLines[tid];
```

IIR Filter

As we know for GPU, time-dependent programming cannot be implemented, which makes it not feasible for IIR filter algorithm. However, even if each of the outputs of a single

RF line from the IIR filter is related to its previous outputs, there is no causal relationship between every two RF lines. So we can parallelize the calculations for the 462 RF lines.

In our LHM experiment system, the total number of RF lines was 462. So we assigned 462 blocks to 462 RF lines and each block was in charge of calculating one single RF line. There was only one thread active in each block since the process of IIR filter was in series so that all the elements within one RF signal should be calculated in one thread. Then we allocated device memory and transferred the data from host memory to device memory. Furthermore, the whole RF line would be too large for GPU, so for each RF line, we only transfer the part of it which we want to filter. Part of the code is shown below (for the full version of code, please refer to Appendix 2.2)

```
int numThreads = 1;
int numBlocks = 462;
cudaMalloc((U16**)&d_data, Nlines*filterLen*sizeof(U16));
for(int i =0;i<Nlines; i++)
{
cudaMemcpy(&d_data[i*filterLen],&data[i*DEPTHPOS+filterStart],
filterLen*sizeof(U16), cudaMemcpyHostToDevice);
}
```

Inside single kernel launch:

First, we calculated IIR filter algorithm in global memory.

We used for loop to calculate each RF line and the threads were controlled by the index 'id'.

```
for (int j=2; j<rfLen; j++)
{
    p[0]=temp[1];
    p[1]=d_data[bid*rfLen+j];// *ELEM(rfLines,i,j);
    d_data[id*rfLen+j]=a0*(p[1])+a1*temp[1]+a2*temp[0]+b1*d_data[id*rfLen+j-1]+b
    2*d_data[id*rfLen+j-2];//
    temp[0]=p[0];
    temp[1]=p[1];
}
```

The total time consumption was 0.568s which was larger than the time for CPU. In order to speed-up, we unrolled the for loop 8 times. So the time decreased to 0.548s.

Then we tried to transfer input data from global memory to shared memory on GPU. The data was processed in shared memory and then transferred the output back to global memory and finally to host memory on CPU. By this way, only one thread in each block will not work well for data transfer. So we increase the number of threads to 512 in each block and every thread was in charge of transferring 7 elements in RF signal.

```
int numThreads = 512;
int numBlocks = 462;
```

Inside the kernel launch part:

We define shared memory as s_data[3584] whose length was big enough to store RF signal.

```
__shared__ float s_data[3584];
```

The data transfer code was shown below:

```
__shared__ int pointsperthread;
pointsperthread = 7;
for(k=0;k<pointsperthread;k++)
{
    if(tid*pointsperthread+k>rfLen)
        s_data[tid*pointsperthread+k]=0;
    else
        s_data[tid*pointsperthread+k]=d_data[bid*rfLen+tid*pointsperthread+k];
}
__syncthreads();
```

After transferring data, thread 0 in each block started the IIR algorithm part same as it in global memory. Then the result was transferred back to global memory.

```
for(k=0;k<pointsperthread;k++)
{
    if(tid*pointsperthread+k<rfLen)
        d_data[bid*rfLen+tid*pointsperthread+k]=s_data[tid*pointsperthread+k];
}
```

However, under this condition, the time consumption rose greatly to 1.16s which means the memory transfer part was inefficient.

Results:

FIR

Table 11 shows the time that they cost to filter the RF signals in 3 different conditions: IIR in CPU, FIR in CPU and FIR in GPU. We calculated the average time consumption for each case which was 0.126s, 0.0854s and 0.0393s, respectively. So the speed up of FIR in CPU and FIR in GPU were 1.48 and 3.20 respectively comparing with the one of IIR in CPU.

	IIR_CPU (s)	FIR_CPU (s)	FIR_GPU (s)
average	0.126	0.0854	0.0393
Speed up	NAN	1.48	3.20

Table 11.Testing result of IIR in CPU, FIR in CPU and FIR in GPU.

IIR

Table 12 shows the time consumption of each condition (CPU, GPU in global memory, GPU in global memory but unrolled and GPU in shared memory).

Time consumption conclusion				
CPU	GPU (global)	GPU (unrolled)	GPU (shared)	GPU (Parallel cross-correlation)
0.226s	0.568s	0.548s	1.16s	0.145s

Table 12. Time consumption under different conditions.

7.4 Discussion

For the result of FIR filter (Table 11), we achieved a 3 times speedup which proved that GPU could give out a better and more efficient performance after parallelization.

However, after we parallelized the IIR program in GPU, we expected a much faster execution. However, compared time consumption in different situations through Table 12, we noticed that the time increased in GPU calculation. This may be caused by two main reasons:

The first reason is the big overhead of transferring data either from host memory to global memory or from global memory to device memory. The second reason is the different clock speeds between CPU (3.2GHz) and GPU (700-800MHz) which means the capability of the single processor on CPU is 3 times stronger than GPU. So while calculating the for loop, the efficiency in GPU was very low.

Further modification will be discussed in the future work part (8.2.3).

8. Conclusion

8.1 Summary

This thesis has implemented a system to measure displacements using a new ultrasound imaging technique—localized harmonic motion imaging. The methods of how we tracked displacements and results of detection have been discussed in the thesis.

First, required background information is briefly covered by this thesis. The introductions of the related technologies for proposed technique such as elastography, vibro-acoustography and acoustic radiation force imaging are provided. Information for the proposed technique such as theory and setup description is also covered in this thesis. From this information, possible problems that this thesis addresses are identified—noise contained in signals.

Then the development of solving the problem has been covered in two ways: analog filters and digital filters. The testing results showed the best combination was low-pass filter, notch filters at 5th, 7th, 8th, 9th and IIR digital filter.

Finally, experimental results with filters introduced by this thesis have shown that the proposed system is feasible for LHM displacement detection. Different LHM amplitudes from samples with different stiffness have also been measured in this thesis. The amplitude decreased with a higher stiffness.

8.2 Future Work

8.2.1 Adding Fundamental Notch Filter (Band-pass Filter)

As was previously covered in Chapter 3, 4 and 5, the high-pass filter didn't work as expected so that we have to remove it from the system which brought a big influence to the acquired results. Because of lacking of this high-pass filter, the difficulty of detecting LHM displacement increased under the interference of low frequency harmonics. So in order to get a better result, it is necessary to build several notch filters for the first 4 harmonic peaks (especially the fundamental peak) or a better band-pass filter to eliminate the low harmonic noise.

8.2.2 Obtain LHM Imaging

Right now the experiment setup is only used for LHM displacement detection for single point. And as was discussed in Chapter 5 and 6, the amplitudes of displacement reflect the stiffness of the sample. So due to this acoustic characteristic of the sample, the long term goal as presented by this thesis is to implement the whole setup for LHM scanning to obtain LHM image according to different stiffness. This may include two main aspects:

(1). Scan of samples with inclusions: To create inclusions within the phantom, we can introduce cylindrical glass rods at the center of the mold before pouring the mix and the rods

extracted after vulcanization. The stiffness of the inclusion should be different from surroundings. The transducers can be mounted to a motor in order to move and control the position of transducers. And at each position, we will repeat the same process as we did for single point displacement detection. Then we can calculate the amplitudes of displacement and compare them from different positions. Finally, we can match the value of amplitudes to grayscale. So by this way, the whole LHM image for samples with inclusions can be achieved based on their different Young modulus [36].

(2). Scan of tissues: Scan of samples with inclusions is just a simulation of real tissue. The ultimate goal of LHM experiment is to obtain the images of real tissues. The procedure of imaging should be exactly the same except different targets.

8.2.3 Program Optimization

Since the IIR filter is recursive algorithm, it is very difficult to be parallelized. And the effect of paralleling the 462 RF signals didn't give out a satisfied result either because of the big overhead or the calculation capability of GPU processors. One suggestion is to combine this part with adding new filters discussed in 8.2.1. After adding new filters which could get rid of the low frequency harmonics, the FIR filter may work well for the data with lower noise and be parallelized to optimize the program.

8.2.4 Mechanical/Acoustical noise sources

As discussed in Chapter 5.3.2.2, the reflection from the interface of water and air or water and plastic holder could bring mechanical/acoustical noise. In order to decrease this noise to minimum, we can add absorbents to cover those interfaces so that the interference noise will be absorbed. Changing the casing of the imaging transducer to an absorbent kind (not metal) should also help improving this noise.

References:

[1]. Sehgal, C.M., Weinstein, S.P., Arger, P.H. & Conant, E.F. 2006, "A review of breast ultrasound", *Journal of mammary gland biology and neoplasia,* vol. 11, no. 2, pp. 113-123.

[2]. Kumar, K., Andrews, M., Jayashankar, V., Mishra, A. & Suresh, S. 2009, "Improvement in diagnosis of breast tumour using ultrasound elastography and echography: A phantom based analysis", *Biomedical imaging and intervention journal,* vol. 5, no. 4, pp. e30.

[3]. J. Jaros, "Ultrasound elastography," Seminar LUT2, University of Kuopio, Finland, Dep. Of Biomedical Engineering, FEEC, BUT Brno, CZ.

[4]. Konofagou, E.E. & Hynynen, K. 2003, "Localized harmonic motion imaging: theory, simulations and experiments", *Ultrasound in medicine & biology,* vol. 29, no. 10, pp. 1405-1413.

[5]. Ophir, J., Alam, S.K., Garra, B., Kallel, F., Konofagou, E., Krouskop, T. & Varghese, T. 1999, "Elastography: ultrasonic estimation and imaging of the elastic properties of tissues", *Proceedings of the Institution of Mechanical Engineers.Part H, Journal of engineering in medicine,* vol. 213, no. 3, pp. 203-233

[6]. Compiled by B. Ahmed, "Vibro-acoustography and its applications," http://ric.uthscsa.edu/personalpages/lancaster/DI2_Projects_2004/AB_Project.pdf.

[7]. Ling, T., Jin, Q., Yao, H. & Deng H 2010, "Preparation and Characterization of a New Ultrasound Tissue-mimicking Phantom", *Bulletin of Advanced Technology Research,* vol. 4, pp. 29.

[8]. Konofagou, E.E., Ophir, J., Krouskop, T. & Garra, B. 2003, "Elastography: From Theory To Clinical Applications", June 25-29, pp. 0367.

[9]. Ophir, J., Garra, B., Kallel, F., Konofagou, E.E., Krouskop, T., Righetti, R. & Varghese, T. 2000, "Elastographic imaging", *Ultrasound in Medicine and Biology,* vol. 26, no. Suppl. 1, pp. 23-29.

[10]. McLauhlin, J. & Renzi, D. 2006, "Shear wave speed recovery in transient elastography and supersonic imaging using propagating fronts", *Institute Physics Publishing,* vol. 22, no. 2, pp. 681-706.

[11]. Heldmuth, L.O., Jean-Luc, G. & Mickael Tanter 2010, "Quantitative stress imaging by combining static and dynamic elastography", in Paris.

[12]. Manduca, A., Oliphant, T.E., Dresner, M.A., Mahowald, J.L., Kruse, S.A., Amromin, E., Felmlee, J.P., Greenleaf, J.F. & Ehman, R.L. 2001, "Magnetic resonance elastography: non-invasive mapping of tissue elasticity", *Medical image analysis,* vol. 5, no. 4, pp. 237-254.

[13]. Heikkila, J. & Hynynen, K. 2006, "Investigation of optimal method for inducing harmonic motion in tissue using a linear ultrasound phased array--a simulation study", *Ultrasonic imaging,* vol. 28, no. 2, pp. 97-113.

[14]. Bishop, J., Samani, A., Sciarretta, J. & Plewes, D.B. 2000, "Two-dimensional MR elastography with linear inversion reconstruction: methodology and noise analysis", *Physics in Medicine and Biology,* vol. 45, no. 8, pp. 2081-2091.

[15]. Ralph, S., Katja, S., Tanja, X., Mickael, T., Claus, C. & Mathias, F. (eds) 2007, *MR elastography of breast lesions: Understanding the solid/liquid duality can improve the specificity of contrast-enhanced MR mammography*.

[16]. Maynard, J. 1996, "Resonant ultrasound spectroscopy", *Physics Today,* vol. 49, pp. 26-31.

[17]. Fatemi, M. & Greenleaf, J.F. 1999, "Vibro-acoustography: an imaging modality based on ultrasound-stimulated acoustic emission", *Proceedings of the National Academy of Sciences of the United States of America,* vol. 96, no. 12, pp. 6603-6608.

[18]. Alizad, A., Fatemi, M., Whaley, D.H. & Greenleaf, J.F. 2004, "Application of vibro-acoustography for detection of calcified arteries in breast tissue", *Journal of ultrasound in medicine : official journal of the American Institute of Ultrasound in Medicine,* vol. 23, no. 2, pp. 267-273.

[19]. Fatemi, M., Wold, L.E., Alizad, A. & Greenleaf, J.F. 2002, "Vibro-acoustic tissue mammography", *IEEE Transactions on Medical Imaging,* vol. 21, no. 1, pp. 1-8.

[20]. Nightingale, K.R., Palmeri, M.L., Nightingale, R.W. & Trahey, G.E. 2001, "On the feasibility of remote palpation using acoustic radiation force", *The Journal of the Acoustical Society of America,* vol. 110, no. 1, pp. 625-634.

[21]. Nightingale, K., Palmeri, M. & Trahey, G. 2006, "Analysis of contrast in images generated with transient acoustic radiation force", *Ultrasound in medicine & biology,* vol. 32, no. 1, pp. 61-72.

[22]. Palmeri, M.L., Frinkley, K.D., Zhai, L., Gottfried, M., Bentley, R.C., Ludwig, K. & Nightingale, K.R. 2005, "Acoustic radiation force impulse (ARFI) imaging of the gastrointestinal tract", *Ultrasonic imaging,* vol. 27, no. 2, pp. 75-88.

[23]. Heikkila, J., Curiel, L. & Hynynen, K. 2010, "Local Harmonic Motion Monitoring of Focused Ultrasound Surgery—A Simulation Model", *Biomedical Engineering, IEEE Transactions on,* vol. 57, no. 1, pp. 185-193.

[24]. Curiel, L., Rajiv, C. & Hynynen, K. 2009, "In Vivo Monitoring of Focused Ultrasound Surgery Using Local Harmonic Motion", *Ultrasound in medicine biology,* vol. 35, no. 1, pp. 65-78.

[25]. Konofagou, E.E., Ottensmeyer, M., Agabian, S., Dawson, S.L. & Hynynen, K. 2004, "Estimating localized oscillatory tissue motion for assessment of the underlying mechanical modulus", *Ultrasonics,* vol. 42, no. 1-9, pp. 951-956.

[26]. Maleke, C. & Konofagou, E.E. 2008, "Harmonic motion imaging for focused ultrasound (HMIFU): a fully integrated technique for sonication and monitoring of thermal ablation in tissues", *Physics in Medicine and Biology,* vol. 53, no. 6, pp. 1773-1793.

[27]. Curiel, L., Huang, Y., Vykhodtseva, N. & Hynynen, K. 2009, "Focused ultrasound treatment of VX2 tumors controlled by local harmonic motion", *Physics in Medicine and Biology,* vol. 54, no. 11, pp. 3405-3419.

[28]. Konofagou, E.E., Ophir, J., Krouskop, T. & Garra, B. 2003, "Elastography: From Theory To Clinical Applications", in Sonesta Beach Resort in Key Biscayne, Florida, June 25-29, pp. 0367.

[29]. Park, E. & Maniatty, A.M. 2006, "Shear modulus reconstruction in dynamic elastography: time harmonic case", *Physics in Medicine and Biology,* vol. 51, no. 15, pp. 3697-3721.

[30]. Maleke, C., Pernot, M. & Konofagou, E.E. 2006, "Single-element focused ultrasound transducer method for harmonic motion imaging", *Ultrasonic imaging,* vol. 28, no. 3, pp. 144-158.

[31]. K. Lacanett , "A basic introduction to Filters⸺Active, passive, and switched capacitor. National semiconductor," 2010 National Semiconductor Corporation, Application Note 779, Texas Instruments.

[32]. Thede, L.D. 2005, *Practical analog and digital filter design,* Artech House, Boston, MA.

[33]. H. Zumbahle, "Using the analog devices active filter design tool," Application Notes 649, Analog Devices, http://www.analog.com/static/imported-files/application_notes/447952852AN-649_0.pdf .

[34]. Rorabaugh, C.B. 1993, *Digital Filter Designer's Handbook: Featuring C Routines*.

[35]. Haykin, S.S. c1991, *Adaptive filter theory,* 2nd edn, Prentice-Hall, Englewood Cliffs, N.J.

[36]. Curiel, L. & Hynynen, K. 2011, "Localized harmonic motion imaging for focused ultrasound surgery targeting", *Ultrasound in medicine & biology,* vol. 37, no. 8, pp. 1230-1239.

[37]. Siberstein, M., Schuster, A., Geiger, D., Patney, A. & Owens, J.D. 2007, *Efficient Computation of Sum-products on GPUs Through Software-Managed Cache*.

[38]. *NVIDIA CUDA C Programming Guide* 2011, CUDA C Programming Guide Version 4.0 edn.

Appendix:

1. MATLAB:

1.1 IIR

(1). IIR_FILTER.m
```
close all;
fs=125e6;
fc=1.485e6;
BW=0.03e6;
har=5;
f=fopen('S4060I2080_2inc5mmO___O_4945.dat','rb');
Continua=fread(f,'uint16');
PerAcquiredLine=reshape(Continua,15000,462);
PerAcquiredLine=PerAcquiredLine';
PerCycle=reshape(Continua,15000,66,7);PerCycle=permute(PerCycle,[3 2 1]);

%% spectrum signal with and without noise
data=squeeze(PerCycle(2,33,:))';
data=data-mean(data);
%getting dc out
data1=FilterNotchO(data,fc,fs,BW,har);
```

(2). FilterNotchO.m
```
function esignal=FilterNotchO(signal,fc,fs,BW,harmonics)

esignal=signal;
figure;
for k=1:harmonics
    esignal=mynotch(esignal,k*fc,fs,BW);
end
```

(3). mynotch.m
```
function out=mynotch(x,fc,fs,BW)

f=fc/fs;
BWc=BW/fs;
R=1-3*BWc;%%radius
K=(1-2*R*cos(2*pi*f)+R^2)/(2-2*cos(2*pi*f));
a0=K;
a1=-2*K*cos(2*pi*f);
```

```matlab
a2=K;
b1=2*R*cos(2*pi*f);
b2=-R^2;

out=x;
for k=3:length(x)
    out(:,k)=a0*x(:,k)+a1*x(:,k-1)+a2*x(:,k-2)+b1*out(:,k-1)+b2*out(:,k-2);
end
a=[a0,a1,a2];
b=[b1,b2];
```

1.2 FIR

(1). FIR_FILTER.m
```matlab
fs=125e6;
fc=1.1575e6;
BW=1e5;
har=5;
a=1e-6*fs;
a=round(a);
b=120e-6*fs;
b=round(b);
timex=[0:1/fs:(15000-1)/fs];

f=fopen('D:\Ping\new\study\project\seminar\data_acquired_with_program\12-20
-2011\AmpTxPRSwFilt_001.dat','rb');
Continua=fread(f,'uint16');
PerAcquiredLine=reshape(Continua,15000,462);
PerAcquiredLine=PerAcquiredLine';
PerCycle=reshape(Continua,15000,66,7);PerCycle=permute(PerCycle,[3 2 1]);
data=squeeze(PerCycle(2,33,:))';
data=data-mean(data);
figure;
plot(timex,data);

n=1;
for k=1:5;
    Wn=[k*(fc*2-BW)/fs k*(fc*2+BW)/fs];
    b = fir1(400,Wn,'stop');
    fname=sprintf('D:\\Ping\\new\\study\\project\\seminar\\coefficient%d.tx
    t',n);
    q=fopen(fname,'wt');
```

```
    fprintf(q,'%d\n',b);
    fclose(q);
    n=n+1;
    data=filter(b,1,data);
end
```

1.3 Adaptive

Adaptive_filter.m
```
close all;
fs=125e6;
fc=1.5e6;
BW=1e6;
u=0.9;

f=fopen('C:\Users\Laura\Desktop\12-22-2011\SSA\\AmpTxPRSwFilt_LPN5_001.dat'
,'rb');
Continua=fread(f,'uint16');
plot(Continua);

a=1e-6*fs;
a=round(a);
b=120e-6*fs;
b=round(b);
timex=[0:1/fs:(15000-1)/fs];
data=(data-8192)/8192;
PerAcquiredLine=reshape(Continua,15000,462);
PerAcquiredLine=PerAcquiredLine';
PerCycle=reshape(Continua,15000,66,7);PerCycle=permute(PerCycle,[3 2 1]);
data=squeeze(PerCycle(2,33,:))';
data=data-mean(data);
data1=Continua(1:15000); % data without noise
data1=data1-mean(data1);


N=length(data);
n=linspace(0,N,N);

w1=0.1;
w2=0.1;
e=zeros(1,N);
figure;
```

```matlab
plot(data)
title('data before filter')
data2=data;

x1=cos(2*pi*60*n);              %signal 1
x2=sin(2*pi*60*n);              %signal 2

%LMS algorithm
for i=1:N
    data(i)=w1*x1(i)+w2*x2(i);
    e(i)=data1(i)-data(i);
    w1=w1+2*u*e(i)*x1(i);
    w2=w2+2*u*e(i)*x2(i);       %update the weight
end

figure;
plot(timex,e)
title('error');
figure;
plot(timex,data)
title('data after filter');
```

2. VC++:

2.1. FIR

CPU code:

```
(1). void CLHMProcessor::AllocateDeviceMemory(unsigned int size){
    // allocate device memory
    cudaError status;

#ifdef _DEBUG
    CString serr;
    serr.Format(_T("\nStarted to Allocate Device Memory\n"));
    OutputDebugString(serr);
#endif

    status = cudaMalloc( (void**)&d_coeficients, FILTERHAR*coefNumber*sizeof(float) );
    if (status != cudaSuccess) {exit(-1);}
    status = cudaMalloc( (void**) &d_rfLinesTemp, (size+500)*sizeof(float) );
    if (status != cudaSuccess) {exit(-1);}
    status = cudaMalloc( (void**) &d_rfLines, (size+500)*sizeof(U16) );
    if (status != cudaSuccess) {exit(-1);}
    status = cudaMalloc( (void**) &d_res, 4*sizeof(float) );
    if (status != cudaSuccess) {exit(-1);}
    cudaMemset(d_res,0,4*sizeof(float));

    status = cudaMalloc( (void**) &d_K, FILTERHAR*sizeof(float) );
    if (status != cudaSuccess) {exit(-1);}
    status = cudaMalloc( (void**) &d_a1, FILTERHAR*sizeof(float) );
    if (status != cudaSuccess) {exit(-1);}
    status = cudaMalloc( (void**) &d_b1, FILTERHAR*sizeof(float) );
    if (status != cudaSuccess) {exit(-1);}
    status = cudaMalloc( (void**) &d_b2, FILTERHAR*sizeof(float));
    if (status != cudaSuccess) {exit(-1);}

#ifdef _DEBUG
    serr.Format(_T("\nDevice Memory Allocated\n"));
    OutputDebugString(serr);
#endif

}
```

```
(2). void CLHMProcessor::FreeDeviceMemory(){
    if(useCUDA)
    {
        // free device memory
#ifdef _DEBUG
        CString serr;
        serr.Format(_T("\nStarted to Free Device Memory\n"));
        OutputDebugString(serr);
#endif
        cudaFree(d_coeficients);
        cudaFree(d_rfLines);
        cudaFree(d_rfLinesTemp);
        cudaFree(d_res);
        cudaFree(d_K);
        cudaFree(d_a1);
        cudaFree(d_b1);
        cudaFree(d_b2);

        cudaFree(c_sgnu1);
        cudaFree(c_sgnu2);
        cudaFree(c_yy);

#ifdef _DEBUG
        serr.Format(_T("\nDevice Memory Freed\n"));
        OutputDebugString(serr);
#endif
    }
}


(3). void CLHMProcessor::Fir_FilterNotchOCUDA(U16* rfLines, int rfLen, double fc, double
fs, double BW, int har)
{
    int numThreads = 512;
    int numBlocks = ceil((float)rfLen/numThreads);
    int coefNum =400;
    U16 cn[5];
    cn[0]= cn[1]= cn[2]= cn[3]=cn[4]= 400;
    U16 *d_cn;
    float * d_coefficients;

    readcoefficients();
    //Copy data to device memory
```

```
    cudaMalloc( (void**)&d_coefficients, coefNum*sizeof(float) );
    cudaMalloc( (void**)&d_cn, har*sizeof(U16) );

    cudaMemcpy(d_rfLines, rfLines, rfLen*sizeof(S16), cudaMemcpyHostToDevice);
    cudaMemcpy(d_coefficients,coefficients,coefNum*sizeof(float),cudaMemcpyHostToDevic
e);
    cudaMemcpy(d_cn, cn, har*sizeof(U16), cudaMemcpyHostToDevice);

    cudaError_t err = cudaGetLastError();
    if( cudaSuccess != err) {
        printf("Memcopy H2D Error: %s",cudaGetErrorString(err));
        FreeDeviceMemory();
        exit(-1);
    }
    //SINGLE KERNEL CALL
    firNotchFilter<<<numBlocks,numThreads>>>(d_rfLines,      d_coefficients,      rfLen,
har,coefNum, d_cn);
    cudaThreadSynchronize();

    err = cudaGetLastError();

    if( cudaSuccess != err) {
        printf("Kernel Error: %s",cudaGetErrorString(err));
        FreeDeviceMemory();
        exit(-1);
    }
    //Copy result from device memory to host memory
    cudaMemcpy( rfLines, d_rfLines, rfLen*sizeof(S16),cudaMemcpyDeviceToHost);

    err = cudaGetLastError();
    if( cudaSuccess != err) {
        printf("Memcopy D2H Error: %s",cudaGetErrorString(err));
        FreeDeviceMemory();
        exit(-1);
    }
    cudaFree(d_coefficients);
    cudaFree(d_cn);
}

(4). void CLHMProcessor::readcoefficients()
{
    int i,j;
```

```
        ifstream File;
        File.open("C:\\Users\\Laura\\Desktop\\seminar\\coefficient1.txt");
        for (i=0; i<5; i++)
        {
            if (File.is_open())
            {
                for(j=0;j<500;j++)
                {
                    File>>coefficients[j];
                }
                File.close();
            }
        }
}
```

GPU Code:

```
(1). __global__ void
firNotchFilter(U16* rfLines, float * coeficients, int rfLen, int har, int coefNum, U16*
cn){

    int id = blockIdx.x*blockDim.x+threadIdx.x;
    int tid = threadIdx.x;
    char blocks = gridDim.x;
    int harmonicDisp;
    int coefDisp=0;

    __shared__ float stmp_rfLines[512];
    __shared__ float s_coeficients[500];

    //Take data from global memory to shared memory
    stmp_rfLines[tid]=0;

    __syncthreads();

    for(int i=0; i<har; i++)
    {
        coefDisp =0;
        //for(int j=0; j<i; j++){
        //  coefDisp+=cn[j];
        //}
        //Put coeficients in shared memory first
```

88

```
        if(tid<cn[i])
            s_coeficients[tid]=coeficients[coefDisp+tid];


        __syncthreads();



        if(id<cn[i]){
            stmp_rfLines[id]=32500;//rfLines[id];
        }
        else if (id>=cn[i] && id< rfLen)
        {
            for(int j=0; j<cn[i]; j++){
                stmp_rfLines[tid]+=s_coeficients[coefDisp+j]*rfLines[id-j];
            }
        }
    }
    __syncthreads();


    rfLines[id]=stmp_rfLines[tid];


    __syncthreads();


}
```

2.2 IIR:

CPU Code:

```
(1). void CLHMProcessor::ProcessMemoryOffCUDA(U16 *data, double *currentDisp, double
*currentLHMAmp, bool *ampUpdated, bool *abort, unsigned int size)
{
    int numThreads = 512;
    int numBlocks = 462;
    int Nlines = 462;
    double corr;
    U16 *data_new;
    U16 *d_data;
    float *d_currentDisp;
    U16 *postRF, *preRF;

    IntTime(data, &intTime, &focusTime, &idxFocus);
```

```
    filterStart = idxFocus - 3000; // window the filter (after the first one)
    if (filterStart<0) filterStart = 0;
    filterLen = (idxFocus+sDelayP+N+100) - filterStart; // don't need to filter much beyond
cross-corr range
    if (filterLen > size) filterLen = size;
    LoadHilbert(data, idxHilb);
    double hilb = Hilbert(); // get max amplitude of rf-line envelope around focus
    if (hilb<1e4) // sets the threshold for echogenicity increase check
        hilbThresh = 1e4;
    else
        hilbThresh = 2e4; // I haven't seen anything legit into 3e4
    *abort = false;
    firstLine = false;
    *currentDisp=0;
    idxDisp=1;
    disps[1]=0;
    *ampUpdated = false;

    //// Rest lines///////
    preRF = (U16 *) malloc(filterLen*sizeof(U16));
    postRF = (U16 *) malloc(filterLen*sizeof(U16));
    data_new = (U16 *) malloc(Nlines*filterLen*sizeof(U16));

    cudaMalloc((U16**)&d_data, Nlines*filterLen*sizeof(U16));
    //cudaMalloc((float**)&d_currentDisp, (Nlines-1)*sizeof(float));

    cudaError_t err = cudaGetLastError();
    if( cudaSuccess != err) {
        printf("Memcopy H2D Error: %s",cudaGetErrorString(err));
        FreeDeviceMemory();
        exit(-1);
    }

    for(int i =0;i<Nlines; i++){
        //cudaMemcpy
        cudaMemcpy(&d_data[i*filterLen], &data[i*DEPTHPOS+filterStart],
filterLen*sizeof(U16), cudaMemcpyHostToDevice);
        err = cudaGetLastError();
        if( cudaSuccess != err) {
            printf("Memcopy H2D Error: %s",cudaGetErrorString(err));
            FreeDeviceMemory();
            exit(-1);
```

```
        }
    }


    clock_t start,finish;
            start=clock();
    FilterCUDA<<<numBlocks,numThreads>>>(d_data, filterLen, hdr.fFUS*1e6, 1/STIME,
FILTERBW, FILTERHAR);



    err = cudaGetLastError();

    if( cudaSuccess != err) {
        printf("Kernel Error: %s",cudaGetErrorString(err));
        FreeDeviceMemory();
        exit(-1);
    }
    cudaThreadSynchronize();
    cudaMemcpy( data_new, d_data, Nlines*filterLen*sizeof(U16),cudaMemcpyDeviceToHost);

    finish=clock();
    double duration;
    duration=(double)(finish-start)/CLOCKS_PER_SEC;
    CString serr;
    serr.Format(_T("elapse time1 is: %f seconds\n"),duration);
    OutputDebugString(serr);



    err = cudaGetLastError();
    if( cudaSuccess != err)
    {
        printf("Memcopy D2H Error: %s",cudaGetErrorString(err));
        FreeDeviceMemory();
        exit(-1);
    }
    cudaFree(d_data);

    cudaThreadSynchronize();
    for(int j = 0; j < (Nlines-1); j++)
        {
            memcpy(&preRF[0], &data_new[(j)*filterLen], filterLen*sizeof(U16));
            memcpy(&postRF[0], &data_new[(j+1)*filterLen], filterLen*sizeof(U16));
```

```
            FindSignalCorrB(&preRF[idxFocus-filterStart],
&postRF[idxFocus+sDelayN-filterStart],N,N+sDelayP-sDelayN, &currentDisp[j], &corr);
            currentDisp[j] *= dz; // convert index displacement to distance, assumes
displLoc inside tissue
            currentDisp[j] += disps[idxDisp]; //this is the previous displacement -->
cumsum();
            disps[++idxDisp] = currentDisp[j];
            *currentLHMAmp=2e-6; // initialize to a junk value for when the amplitude is
not updated; maybe not the best solution
            LoadHilbert(postRF,idxHilb-filterStart);
            if (Hilbert() > hilbThresh)
                *abort = true;

            CalcAmplitude(disps, currentLHMAmp);
            *ampUpdated = true;
            idxDisp=0;
            disps[0]=currentDisp[j];
        }
}
```

GPU code:

```
(1). __global__ void
FilterCUDA (U16* d_data, int rfLen, float fc, float fs, float BW, float har)
{
    int id = blockIdx.x*blockDim.x+threadIdx.x;
    int tid = threadIdx.x;
    int k =0;
    __shared__ float f, BWc, R, K, a0, a1, a2, b1, b2;
    float p[2],
    float temp[2];
    __shared__ float s_data[3584];
    __shared__ int pointsperthread;

    pointsperthread = 7;

    for(k=0;k<pointsperthread;k++){
        if(tid*pointsperthread+k>3433)
            s_data[tid*pointsperthread+k]=0;
        else

    s_data[tid*pointsperthread+k]=d_data[blockIdx.x*3433+tid*pointsperthread+k];
```

```
}
__syncthreads();
if(tid==0){
    id = id*rfLen;
    BWc=BW/fs;
    R=1-3*BWc;
    b2=-R*R;
    temp[0]= s_data[id+0];// *ELEM(rfLines,i,0);
    temp[1]= s_data[id+1];// *ELEM(rfLines,i,1);

    for( k=3; k<har; k++)
    {
        f=k*fc*1e6/fs; //frequency to notch out
        K=(1-2*R*__cosf(2*PI*f)+R*R)/(2-2*__cosf(2*PI*f));
        a0=K;
        a1=-2*K*__cosf(2*PI*f);
        a2=K;
        b1=2*R*__cosf(2*PI*f);

            for (int j=2; (j+7)<rfLen; j=j+8) //unrolling 8 times
            {
                p[0]=temp[1];
                p[1]=s_data[id+j];// *ELEM(rfLines,i,j);

s_data[id+j]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j-1]+b2*s_data[id+j-2];
                temp[0]=p[0];
                p[0]=p[1];
                p[1]=s_data[id+j+1];

s_data[id+j+1]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j]+b2*s_data[id+j-1];
                temp[0]=p[0];
                p[0]=p[1];
                p[1]=s_data[id+j+2];

s_data[id+j+2]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j+1]+b2*s_data[id+j];
                temp[0]=p[0];
                p[0]=p[1];
                p[1]=s_data[id+j+3];

s_data[id+j+3]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j+2]+b2*s_data[id+j+1];
                temp[0]=p[0];
                p[0]=p[1];
```

```
                    p[1]=s_data[id+j+4];


s_data[id+j+4]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j+3]+b2*s_data[id+j+2];
                    temp[0]=p[0];
                    p[0]=p[1];
                    p[1]=s_data[id+j+5];


s_data[id+j+5]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j+4]+b2*s_data[id+j+3];
                    temp[0]=p[0];
                    p[0]=p[1];
                    p[1]=s_data[id+j+6];


s_data[id+j+6]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j+5]+b2*s_data[id+j+4];
                    temp[0]=p[0];
                    p[0]=p[1];
                    p[1]=s_data[id+j+7];


s_data[id+j+7]=a0*p[1]+a1*p[0]+a2*temp[0]+b1*s_data[id+j+6]+b2*s_data[id+j+5];
                    temp[0]=p[0];
                    temp[1]=p[1];


                }
            }
        }
    __syncthreads();
    for(k=0;k<pointsperthread;k++){
        if(tid*pointsperthread+k<3433)
            d_data[bid*3433+tid*pointsperthread+k]=s_data[tid*pointsperthread+k];
    }
}
```