A REAL-TIME MONITORING
OF A CLOSED-LOOP WATER COOLING SYSTEM
USING A WIRELESS ACOUSTIC EMISSION SENSOR

Chavadol Mekchar

A Thesis submitted in partial fulfillment of the requirements
for The Degree of Master of Engineering Program in Engineering Technology
Graduate school
Thai-Nichi Institute of Technology
Academic year 2014

Thesis Title	A Real-Time Monitoring of a Closed-Loop Water Cooling System using a Wireless Acoustic Emission Sensor
By	Chavadol Mekchar
Faculty	Engineering Technology
Advisor	Asst. Prof. Dr. Wimol San-um

The Graduate School of Thai-Nichi Institute of Technology has been approved and accepted as partial fulfillment of the requirement for the Master's Degree

…………………………………………..Dean of Graduate school
(Assoc. Prof. Dr. Pichit Sukchareonpong)
Month………… Date………..…….. Year…………..

Thesis Committees

…………………………………………..Chairman
(Assoc. Prof. Dr. Worapong Tangsrirat)

…………………………………………..Committee
(Asst. Prof. Dr.Warakorn Srichavengsup)

…………………………………………..Committee
(Dr. Thitiporn Lertrasdachakul)

…………………………………………..Advisor
(Asst. Prof. Dr. Wimol San-um)

Chavadol Mekchar: A Real-Time Monitoring of a Closed-Loop Water Cooling System using a Wireless Acoustic Emission Sensor. Advisor: Asst. Prof. Dr. Wimol San-um, 129 pp.

This thesis proposes a real-time monitoring of a closed-loop water cooling system using a wireless acoustic emission sensor. The proposed system comprises four major components, i.e. a preamplifier, a microcontroller-based analog-to-digital converter, wireless transmission modules and a graphic user interface. The preamplifier is implemented using BiMOS operational amplifier model CA3140 which provides high input impedance and high speed performance. The signal is subsequently digitized by STM32F4 microcontroller and transmits to the personal computer wirelessly via wireless transmission modules using XBee. The experiments show that the proposed system offers a cost-effective data acquisition system for an AE sensor to monitor a closed-loop water cooling system which can overcome the limitation on a long distance operation in order to utilize in the real world applications.

Graduate school

Field of Engineering Technology

Academic Year 2014

Student’s signature………………………

Advisor’s signature……………………...

Acknowledgements

The author wishes to express his profound gratitude and to respectfully dedicate this work to his parent and family members for their endless encouragements, love and sacrifices. The author is most grateful to his advisor, Asst. Prof. Dr. Wimol San-Um, for his valuable supervision, supports and encouragements throughout the study. Grateful acknowledgements are also made to Asst. Prof. Dr. Wipawadee Wongsuwan, the director of master of engineering technology program for her supports. In addition, grateful acknowledgements are made to Assoc. Prof. Dr. Worapong Tangsrirat, Asst. Prof. Dr. Warakorn Srichavengsup, Dr. Thitiporn Lertrasdachakul, members of thesis committee, for their valuable suggestions and comments. The author also wishes to acknowledge Thai-Nichi Institute of Technology for a great opportunity of master study. The author sincerely appreciates all of his friends in Intelligent Electronic Systems Research laboratory for their friendships and good will.

Chavadol Mekchar

Table of Contents

Table Contents (Continued)

List of Table

List of Figures

Lists of Figures (Continued)

Lists of Figures (Continued)

Lists of Figures (Continued)

Lists of Figures (Continued)

Lists of Figures (Continued)

Chapter 1
Introduction

1.1 Background

The closed-loop water cooling system is developed from methods, which are employed for the cooling in the early engine design processes. In typical closed-loop systems, water is circulated in a closed cycle and subjected to alternate cooling and heating without air contact. Heat will be absorbed by the water in such a closed-loop system. Subsequently, heat is transferred by a water-to-water exchanging process to the recirculating water of an open recirculating system, from which the heat is lost to atmosphere. Figure 1.1 shows the typical closed-loop cooling system, mainly including cooling water, a heat exchanger, a pump, a pipeline, and a load.

Closed-loop water cooling systems are well suited to the cooling of gas engines and compressors. Diesel engines in stationary and locomotive services normally utilize radiator systems in a similar manner to generally-known automobile cooling systems. Other closed-loop water cooling applications include lubricating oil and sample coolers in power plants. Closed systems are also widely used in air conditioning chilled water systems to transfer the refrigerant cooling to air washers, in which the air is chilled.

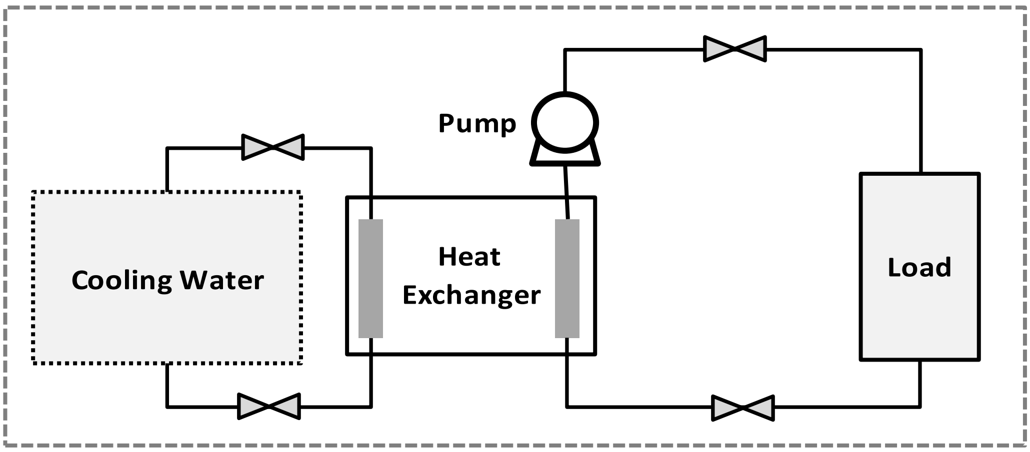


Figure 1.1 A typical closed-loop cooling system.

Closed-loop water cooling systems are also widely used in industries in order to keep machine temperatures in normal condition for increasing machine reliability and extending lifetime. In the case when there is a fault in the system without acknowledge by any operators or maintenance teams, it may cause machines to breakdown and stop production, which is relatively trouble for industries.

In present, the available technology to detect the fault in closed-loop water cooling systems is still complicated, requiring long time for detection as well as high investment cost such as water flow transmitter and water pressure transmitter. Meanwhile, there are researches that apply the acoustic emission technology to detect leaks in pipelines. The Acoustic Emission (AE) signal is a transient elastic wave generated by the rapid release of energy from localized sources within a material. Afterwards, this AE signal is detected by acoustic emission sensors and instruments.

This research therefore studies the application of the AE sensor for a real-time monitoring system of closed-loop water cooling system by detecting the AE signal generated by turbulent flows of cooling water. The detected AE signal is transmitted via wireless equipment to personal computer for signal analyses in both time and frequency domains.

1.2 Motivations

1.2.1 The existing fault diagnosis system for closed-loop water cooling systems is still complicated and expensive. This thesis proposes the cost-effective system to monitor a closed-loop water cooling system in order to prevent major breakdown of machines.

1.2.2 The distance between the AE sensor and the data acquisition system affects both signal quality and cable installation problems. This thesis proposes the cost-effective wireless AE sensor with microcontroller–based data acquisition system in order to overcome the limitation on a long distance operation.

1.3 Objectives

1.3.1 To study the relationship between the AE signal and the faults in closed-loop water cooling system in both time and frequency domains.

1.3.2 To design a real-time monitoring of closed-loop water cooling system using a wireless AE sensor with microcontroller–based data acquisition system.

1.4 Research Scopes

The overall research scope of this thesis is shown in Figure 1.2. The scope focuses on a real-time monitoring of closed-loop water cooling system using wireless AE signal transmission combined with a microcontroller–based data acquisition system and a Graphic User Interface (GUI) in MATLAB. This research also focuses on turbulent flow detections in the closed-loop water cooling system of critical machines under unusual pressure and flow rate conditions. The research scope of this thesis consists of five major parts as follows;

1.4.1 The low-pass filter and pre-amplifier circuits are designed and tested on printed circuit board, focusing on the frequency response at 100 kHz and the gain within 20 dB to interface with an AE sensor.

1.4.2 The Analog-to-Digital (A/D) converter is designed and programmed on STM32F4 microcontroller. This A/D converter converts the AE signals to a personal computer and hence the GUI software. Simulations are performed through Orcad PSpice and a MATLAB Simulink. Signal quality analysis involves frequency response, Total Harmonic Distortion (THD), and Power Spectral Density (PSD).

1.4.3 The wireless AE data transmission system is designed and programmed on XBee wireless module to transmit wirelessly at the frequency of 2.4 GHz.

1.4.4 The GUI with data analysis software is programmed on MATLAB to display AE signal in both time and frequency domains on a personal computer.

1.4.5 The Closed-loop water cooling system is studied by focusing on water pressure in steel pipelines in the range of 5-6 bars.

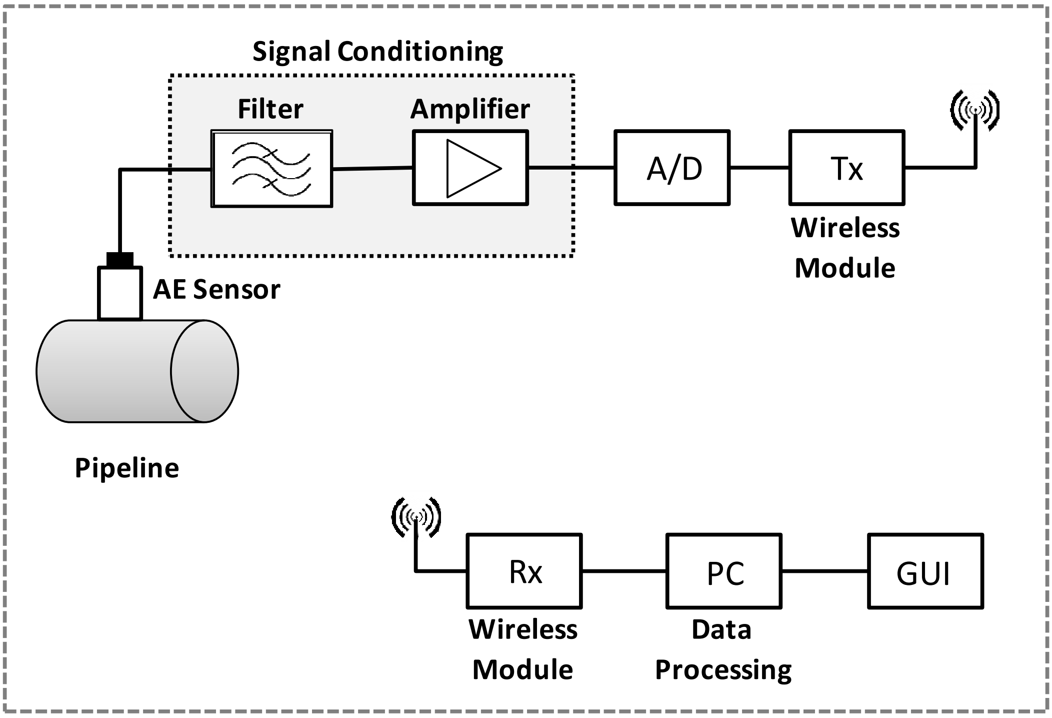


Figure 1.2 The proposed overall research scope.

1.5 Expected Outcomes

1.5.1 To understand the relationship between acoustic emission signal and the fault in a closed-loop water cooling system in both time and frequency domain.

1.5.2 To achieve a real-time monitoring system for a closed-loop water cooling system using wireless acoustic emission sensor with a microcontroller–based data acquisition system.

1.6 Definitions

1.6.1 Acoustic emission (AE) is the sound waves produced when a material undergoes stress (internal change), as a result of an external force. AE is a phenomenon occurring in for instance mechanical loading generating sources of elastic waves. This occurrence is the result of a small surface displacement of a material produced due to stress waves generated when the energy in a material or on its surface is released rapidly. The wave generated by the source is of practical interest in methods used to stimulate and capture AE in a controlled fashion, for study and/or use in inspection, quality control, system feedback, process monitoring and others.

1.6.2 Microcontroller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Program memory in the form of NOR flash or OTP ROM is also often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications. Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes.

1.6.3 Wireless communications is a type of data communication that is performed and delivered wirelessly. This is a broad term that incorporates all procedures and forms of connecting and communicating between two or more devices using a wireless signal through wireless communication technologies and devices. Wireless communication generally works through electromagnetic signals that are broadcast by an enabled device within the air, physical environment or atmosphere. The sending device can be a sender or an intermediate device with the ability to propagate wireless signals. The communication between two devices occurs when the destination or receiving intermediate device captures these signals, creating a wireless communication bridge between the sender and receiver device.

1.6.4 Analog-to-digital converter is any device that converts analog signals (continuous quantity) into digital signals (discrete time digital representation). The analog signal is a continuous sinusoidal wave form that cannot be read by a computer, hence the need for conversion. By converting the analog signal, data can be amplified, added or taken from the original signal. The usual conversion process makes use of a comparator, where at some points, the value of the input analog signal is compared to a standard, so the converter will know if the input warrants a high or low signal. In the case of audio digital conversion, the amplitude or volume is constantly measured, and the output is a list of binary data that contains sound wave values.

1.6.5 MATLAB is advanced computer program (High-level Language) for technical computing that includes numerical computation. Complex graphics And replication to visualize the image is simple and clear name MATLAB stands for Matrix Laboratory original MATLAB program is written to use in the calculation of matrix or a matrix software which MATLAB is a program developed unceasingly. The program is easy to understand. And complex programming When put to use, and can see the results quickly. For this reason it makes MATLAB program has been used extensively in various fields.

Chapter 2
Related Theories and Literature Reviews

2.1 Introduction

This chapter describes the related theories particularly AE principles and its parameters, filter circuit, analog-to-digital converter and wireless communications. The literature reviews are also studied, focusing on the applications, techniques and characteristics of AE sensor for pipelines fault detection in which the knowledge from theories and literature reviews are synthesized and applied for this research.

2.2 Related Theories

2.2.1 Acoustic Emissions

It is commonly known that materials often emit sound under mechanical stress or when internal structures break. Generally speaking, these (often inaudible) breaking sounds are referred to as "Acoustic Emissions" (AE): "AE refers to the generation of transient elastic waves produced by a sudden redistribution of stress in a material. When a structure is subjected to an external stimulus, i.e. change in pressure, load, or temperature, localized sources trigger the release of energy, in the form of stress waves, which propagate to the surface. Sources of AE vary from natural events like earthquakes and rock bursts to the initiation and growth of cracks, slip and dislocation movements, melting, twinning, and phase transformations in metals. In composites, matrix cracking and fiber breakage and debonding contribute to acoustic emissions." [1]

AE sensing technology began to evolve in the middle of the 20^{th} century. It is a so-called non-destructive testing (NDT) method, providing information about ongoing stress processes in the material sample under test. In contrast to other NDT methods, which are mostly applied before or after the material is being stressed, AE sensing captures processes while they are taking place. This makes AE sensing particularly interesting for applications that require constant monitoring of certain structures, e.g. alarm systems for pipeline leaks. Often, this method is used to detect material failure at a very early stage of damage and before the structure fails completely [2].

2.2.1.1 Sensor Technology

AE signals are often being captured with piezoelectric sensors. These sensors_are directly attached to the surface of the material sample under test, converting dynamic motions at the surface into an electrical signal.

When using piezoelectric sensors, it must be considered that they are normally operated at resonance and therefore do not allow broadband detection of AE signals. Nevertheless, the frequency ranges of the expected signals are often fairly well known, making it possible to choose the right sensor before the experiment. Thus, their ease of use as well as their high sensitivity makes piezoelectric sensors the instrument of choice for many AE applications, even though more sophisticated AE sensor concepts exist (e.g. sensors based on laser systems or on fiber optics).

Before A/D conversion, the electrical signal at the output of a piezoelectric sensor normally needs to be filtered and amplified. This process is also known as “signal conditioning”. It is obvious, that these components of an acquisition system are very sensitive and heavily influence the overall measurement quality. The conditioned signal is then sampled and all further processing is then performed digitally.

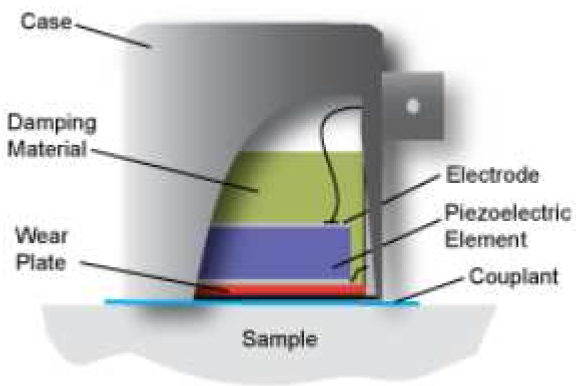


Figure 2.1 The structure of Piezoelectric AE sensor.

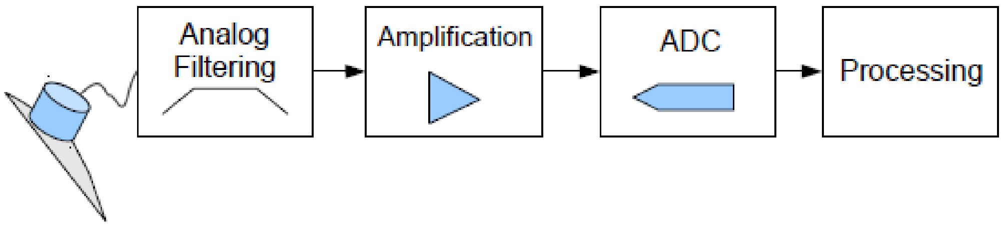


Figure 2.2 The AE data acquisition process.

There are two types of acoustic emissions as defined by ASTM E 1316 [3]: burst type as a qualitative description of the discrete signal related to an individual emission event occurring within the material, and continuous type as a qualitative description of the sustained signal level produced by rapidly occurring acoustic emission sources. Majeed; and Murthy [4] defined the AE signals including nonzero rise time as compared to traditional zero rise time signal presentation. In this study, the burst type AE signal $V_b(t)$ for single frequency f_o is idealized in order to include the arrival time factor into the formulation using the following equation:

$$V_b(t) = V_0 \sin(2\pi f_0 t)\left\{\left(1 - e^{-\frac{t-t_{arrival}}{t_{rise}}}\right) \in 0..1\right\} * e^{-\frac{t-t_{arrival}}{t_{decay}}} H[t - t_{arrival}] \quad (2.1)$$

where the term $\left\{\left(1 - e^{-\frac{t-t_{arrival}}{t_{rise}}}\right) \in 0..1\right\}$ indicates the rise time function normalized to be in the range of 0 to 1, t_{rise} is rise time, the term $e^{-\frac{t}{t_{decay}}}$ indicates the decay time with t_{decay}, the term $H[t - t_{arrival}]$ is Heaviside function indicating the waveform arrival to the sensor at $t_{arrival}$.

The continuous type AE signal $V_{continuous}(t)$ for single frequency form is idealized using the following equation:

$$V_{continuous}(t) = \sin(2\pi f_0 t) \sum_{i=1}^{\infty} V_i \left\{\left(1 - e^{-\frac{t-t_{arrival}}{t_{rise}}}\right) \in 0..1\right\} * e^{-\frac{t-t_{arrival}}{t_{decay}}} H[t - t_{arrival}(i)] \quad (2.2)$$

2.2.1.2 Data Processing and Signal Parameter Analysis

As most modern data analysis systems, today's AE systems almost exclusively perform the necessary signal detection, processing and characterization tasks in the digital domain. AE activity normally occurs rapidly and randomly, resulting in distinct “pulses” of oscillation in the measured AE signal. Conventionally, a threshold is defined by the user in order to distinguish AE activity from background noise. Once the AE signal crosses this threshold, an AE event is recognized. The event is considered to be terminated when the threshold has not been exceeded for a certain, user-defined amount of time (i.e., “hit-lockout time”, “dead time” or “post trigger time”). The AE signal between the first threshold crossing and the end of the post trigger time is also referred to as “AE Hit” or “AE event”. As especially in the early days of AE testing technology, all of the signal processing had to be done in the analog domain, some common signal parameters have evolved that still form the basis for many of today's AE testing and analysis applications. These “classical” AE signal parameters are as follows: 1) Event Count/Hit Count: A signal that exceeds the threshold and causes the AE system to accumulate data. Hit counts are often used to show the overall AE activity over a certain period of time, 2) Length/Duration: The amount of time between the first threshold crossing of an event and the end of the post trigger time, 3) Amplitude: The maximum signal amplitude within an event, 4) Rise Time: The time interval between the first threshold crossing and the time of the peak amplitude, 5) Pulse Count/Count: The number of times an AE event signal exceeds the threshold value and 6) Energy: There is no single agreed definition of AE signal energy. In this thesis, we define the energy of an AE event as the signal energy contained in the event, i.e., the sum of the squared sample amplitudes. Another popular energy definition is the measured area under the rectified signal envelope.

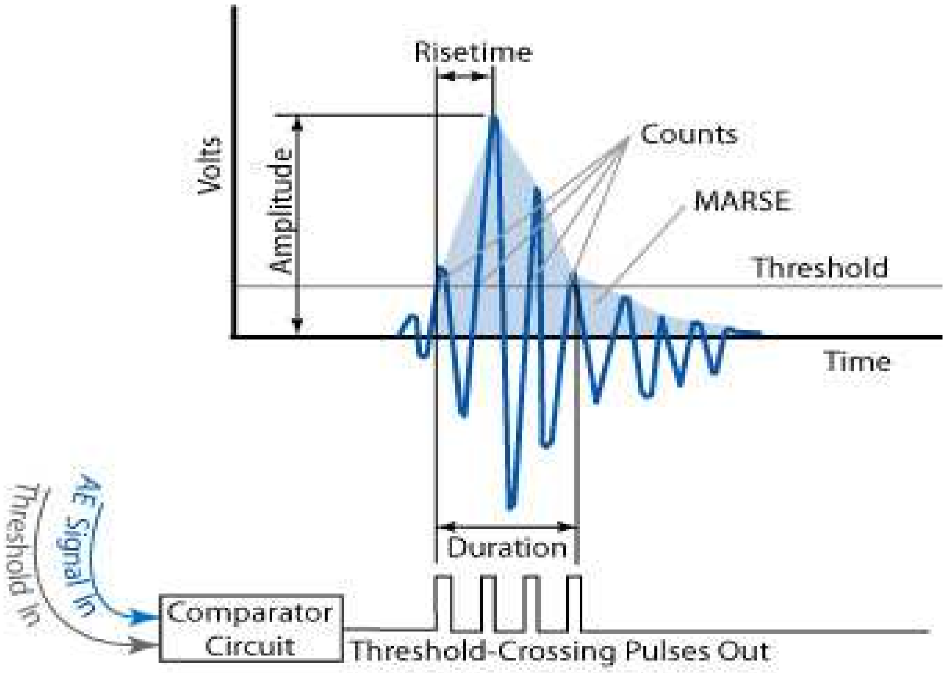


Figure 2.3 Characteristics of AE event parameters

Generally, the continuous acoustic emission signal is applied for leak detection application which the most parameters are average signal level (ASL) and root mean square (AE_{rms}). The below are equations of each parameter.

$$ASLv = \frac{1}{T}\int_{to}^{t0+T} |v(t)dt| = \frac{1}{N}\sum_{n=1}^{N} |v(n)| \qquad (2.3)$$

where ASL_v is an average signal level in voltage and v(t) is an AE signal in voltage. Additionally,

$$AErms = \sqrt{\frac{1}{T}\int_{t0}^{to+T} v^2\,(t)dt} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} v^2\,(n)} \qquad (2.4)$$

where AE_{rms} is a root mean square value of AE signal in voltage and v (t) is an AE signal in voltage.

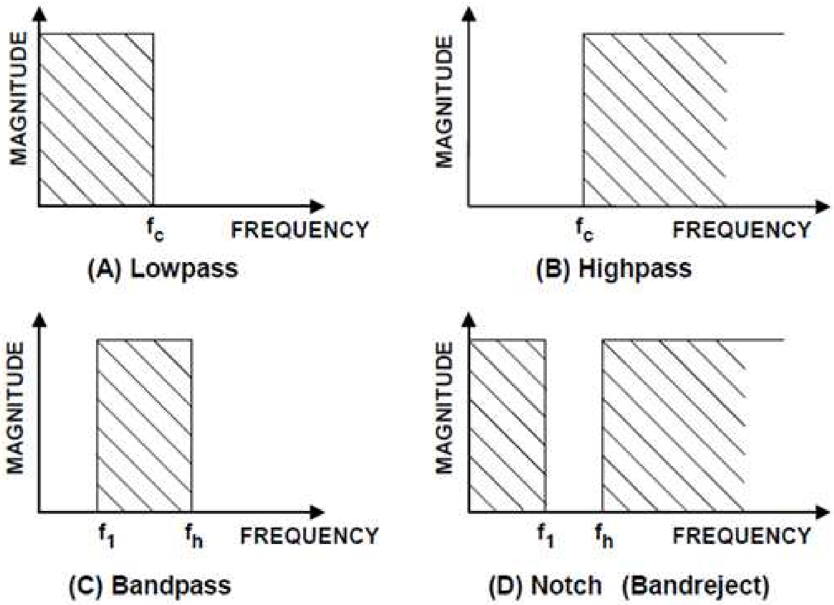


Figure 2.4 The ideal filter.

2.2.2 Filter Circuit

Filters are networks that process signals in a frequency-dependent manner. The basic concept of a filter can be explained by examining the frequency dependent nature of the impedance of capacitors and inductors. Consider a voltage divider where the shunt leg is a reactive impedance. As the frequency is changed, the value of the reactive impedance changes, and the voltage divider ratio changes. This mechanism yields the frequency dependent change in the input/output transfer function that is defined as the frequency response.

An ideal filter will have an amplitude response that is unity (or at a fixed gain) for the frequencies of interest (called the pass band) and zero everywhere else (called the stop band). The frequency at which the response changes from pass band to stop band is referred to as the cut off frequency.

Figure 2.4 (A) shows an idealized low-pass filter. In this filter the low frequencies are in the pass band and the higher frequencies are in the stop band. The functional complement to the low-pass filter is the high-pass filter. Here, the low frequencies are in the stop-band, and the high frequencies are in the pass band.

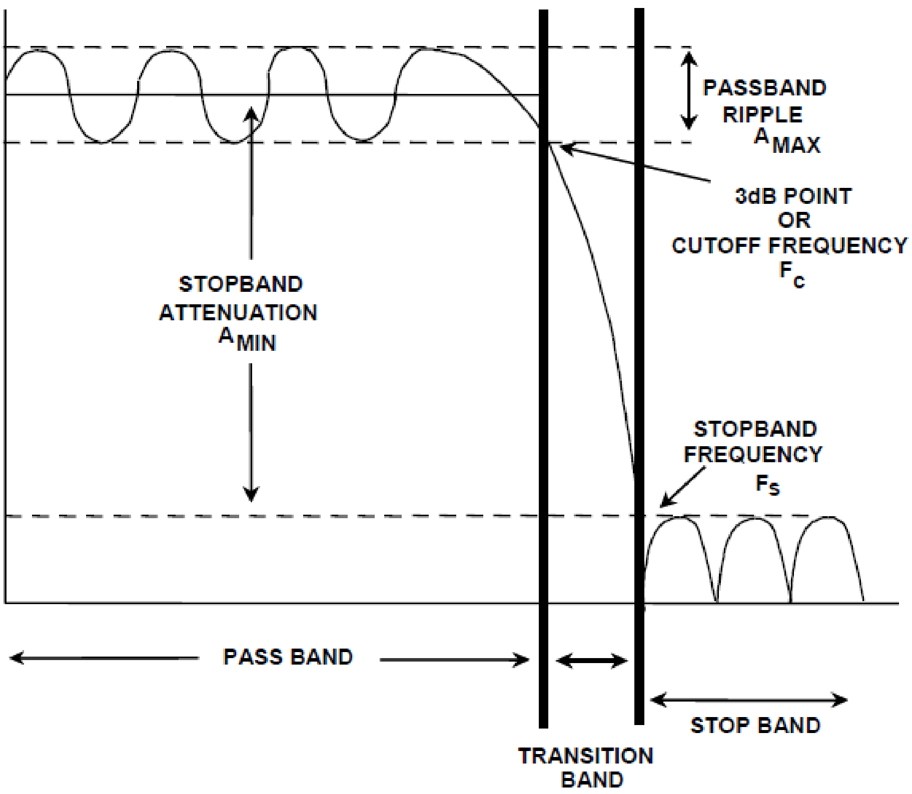


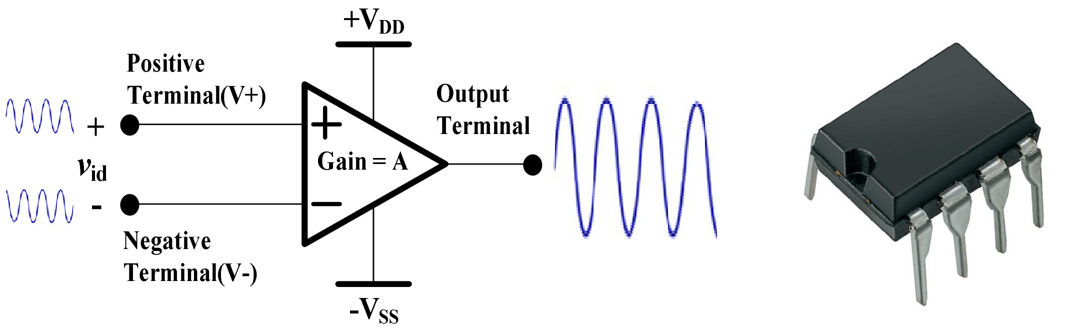
Figure 2.5 Key filter parameters.

Figure 2.4 (B) shows the idealized high-pass filter. If a high-pass filter and a low-pass filter are cascaded, a band pass filter is created. The band pass filter passes a band of frequencies between a lower cutoff frequency, f_l, and an upper cutoff frequency, f_h. Frequencies below f_l and above f_h are in the stop band. An idealized band pass filter is shown in Figure 2.4 (C). A complement to the band pass filter is the band-reject, or notch filter. Here, the pass bands include frequencies below f_l and above f_h. The band from f_l to f_h is in the stop band. Figure 2.4 (D) shows a notch response.

The idealized filters defined above, unfortunately, cannot be easily built. The transition from pass band to stop band will not be instantaneous, but instead there will be a transition region. Stop band attenuation will not be infinite. The five parameters of a practical filter are defined in Figure 2.5, opposite. The cutoff frequency (F_c) is the frequency at which the filter response leaves the error band (or the −3 dB point for a Butterworth response filter). The stop band frequency (F_s) is the frequency at which the minimum attenuation in the stop band is reached. The pass band ripple (A_{max}) is the variation (error band) in the pass band response. The minimum pass band attenuation (A_{min}) defines the minimum signal attenuation within the stop band. The steepness of the filter is defined as the order (M) of the filter. M is also the number of poles in the transfer function. A pole is a root of the denominator of the transfer function. Conversely, a zero is a root of the numerator of the transfer function. Each pole gives a –6 dB/octave or –20 dB/decade response. Each zero gives a +6 dB/octave, or +20 dB/decade response.

2.2.3 Operational Amplifier

An operational amplifier (op-amp) is a DC-coupled high-gain electronic amplifier with a differential input and, generally, a single-ended output. Figure 2.6 shows the standard op-amp symbol and a commercial op-amp package. The op-amp produces an output voltage that is typically hundreds of thousands of times larger than the voltage difference between its input terminals. The op-amp was intended for use with external feedback elements which determine the resultant function or operation, and hence the name “operational amplifier,” denoting that an amplifier can perform a variety of operations. The op-amp has been very popular building blocks for circuit designs since characteristics of an op-amp are set by external components with small dependence on temperature changes or manufacturing variations. Op-amps have been employed in most of electronic and electrical applications such as controls, communications, and signal processing.



(a) A standard Op-Amp symbol **(b) A commercial Op-Amp package**

Figure 2.6 The standard op-amp symbol and a commercial op-amp package.

With reference to Figure 2.6, there are two input terminals, i.e. non-inverting (V-) and inverting (V+) terminals, an output pin (V_O), and two power supply terminal (V_{DD}, V_{SS}). The ideal op-amp has infinite gain for differential input signals. In addition, the ideal op-amp has infinite input impedance and zero output impedance. The dual supply presents the same absolute value of voltage to ground from either side, and consequently the center connection defines the common line or ground potential. The exceptions are AC amplifier circuits that may employ a single power supply. This is accomplished by creating a floating AC ground with DC blocking capacitors. In such circuits, a source of “half-supply” creates a “virtual ground” exactly half way between the positive supply and ground potentials. This chapter deals with the study of ideal op-amp properties. Some useful op-amp configurations with external feedback are investigated.

2.2.3.1 Ideal Amplifier Properties

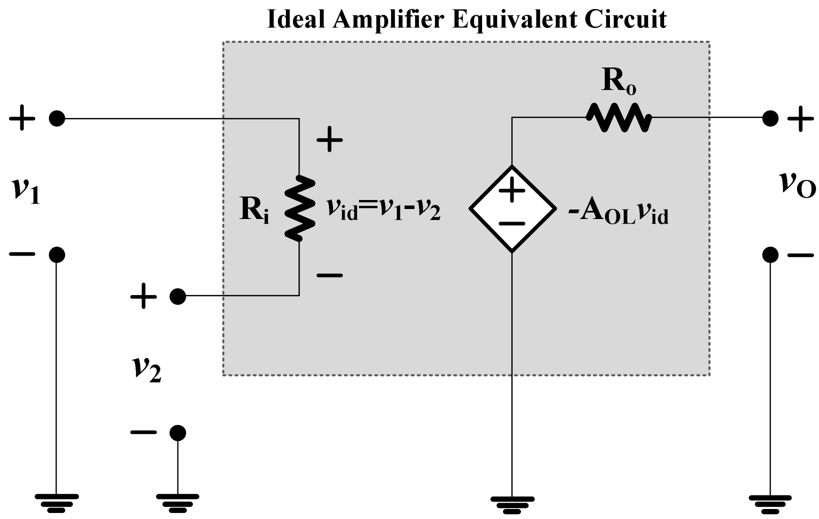


Figure 2.7 The ideal op-amp equivalent circuit.

The ideal op-amp equivalent circuit is shown in Figure 2.7. It is seen that the input resistance R_i is the Thevenin equivalent resistance seen at the input terminals while the output resistance R_o is the Thevenin equivalent resistance seen at the output terminal. In addition, the output section consists of a voltage-controlled source in series with the output resistance R_o. The differential input voltage v_{id} is given by

$$v_{id} = v_{i1} - v_{i2} \tag{2.5}$$

where v_{i1} is the voltage at the inverting terminal and v_{i2} is the voltage at the non-inverting terminal. The op-amp detects the voltage difference between its two input terminals, and multiplies it by the gain A_{OL}, causing the resulting output voltage. Consequently, the output v_o is given by

$$v_O = -A_{OL} v_{id} = -A_{OL}(v_{i1} - v_{i2}) \tag{2.6}$$

The gain A_{OL} is called an open-loop voltage gain as no external feedback from output to input is connected. Ideally, the op-amp has an infinite A_{OL}, an infinite input resistance R_i, and zero output resistance. As a consequence form such ideal characteristics, the currents into both input terminals are zero is due to infinite input resistance. In other words, an infinite resistance between two input terminals implies an open circuit and therefore current cannot enter the op-amp. Consequently, the voltage across the input terminals is negligibly small, i.e. $v_{id}=v_{i1} - v_{i2}=0$. In circuit analysis, it can be assume that v_{i1} equals v_{i2}.

In practice, a practical op-amp has the value of A_{OL} in the range 10^5 to 10^8, the value of R_i in the range $10^6\Omega$ to $10^{13}\Omega$, and the value of R_o in the range 10Ω to 100Ω. As the open-loop gain A_{OL} is very large in the range 10^5 to 10^8, an extremely small difference in the two input voltages causes the op-amp operate in its saturated states, i.e. the output voltage reach its extreme positive or negative voltage limits. As this is seldom desirable, a negative feedback using external resistors is generally applied to reduce the overall gain through signal feedback. A negative feedback is achieved when the output is fed back to input terminals of the op amp. As a result of the negative feedback, the closed-loop gain is almost insensitive to the open-loop gain A_{OL}. For this reason, op amps are used in circuits with feedback paths.

2.2.3.2 Inverting and Non-Inverting Amplifiers

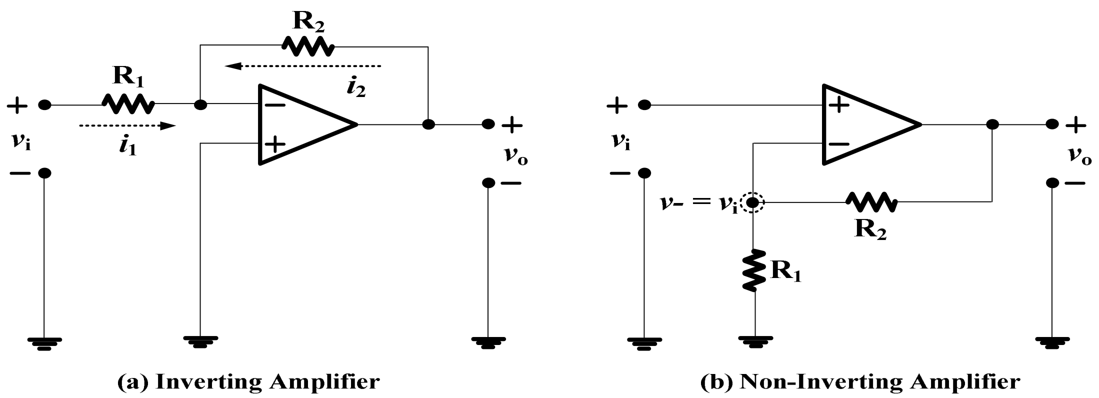


Figure 2.8 Inverting and non-inverting amplifiers.

Two simple configurations of op-amp with feedback are inverting and non-inverting amplifiers as shown in Figure 2.8. The inverting amplifier in Figure 2.8 (a) consists of a single op-amp and two resistors. This inverting amplifier converts positive voltages from the inputs to negative amplified voltages on the output and vice-versa. Based on Kirchhoff's current law, the input current can be found by

$$i_1 = \frac{v_i - v_+}{R_1} = \frac{v_i - 0}{R_1} = \frac{v_i}{R_1} \tag{2.7}$$

As there is no current flows into the input terminals of the ideal op amp, therefore $i_2 = i_1$ and

$$v_o = v_+ - i_2 R_2 = 0 - i_2 R_2 = -i_1 R_2 = -\left(\frac{v_i}{R_1}\right) R_2 = -v_i\left(\frac{R_2}{R_1}\right) \tag{2.8}$$

As a result, the closed-loop gain of the inverting amplifier is given by

$$A_{CL} = \frac{v_o}{v_i} = -\left(\frac{R_2}{R_1}\right) \tag{2.9}$$

The major advantage of the inverting amplifier is that a particular gain can be achieved by changing the values of feedback resistor R_2. However, two significant disadvantages are low input impedance and a dual power supply is required, which is not convenient for modern circuit designs with a single positive supply.

The non-inverting amplifier in Figure 2.8 (b) also consists of a single op-amp and two resistors. The input is directly connected to the positive terminal of the op-amp and the feedback path is between the negative terminal and the output terminal. Based on the ideal property of an op-amp that V+ = V-, the non-inverting amplifier can be analyzed using a simple voltage divider and the closed loop gain is given by

$$v_i = \left(\frac{R_1}{R_1 + R_2}\right) v_o \tag{2.10}$$

Arranging (2.10) for a closed-loop gain yields

$$A_{CL} = \frac{v_o}{v_i} = \left(1 + \frac{R_2}{R_1}\right) \tag{2.11}$$

The non-inverting amplifier provides a voltage gain of greater than one with no signal inversion and can be used as the input stage of a differential instrumentation amplifier. The non-inverting amplifier has very high input impedance and can be implemented with a single 0 to 5V supply. The gain can be controlled by replacing R_2 with a variable resistor.

2.2.4 Data acquisition and conversion systems

Data acquisition and conversion systems are used to acquire analog signals from one or more sources and convert these signals into digital form for analysis or transmission by end devices such as digital computers, recorders, or communications networks. The analog signal inputs to data acquisition systems are most often generated from sensors and transducers which convert real-world parameters such as pressure, temperature, stress or strain, flow, etc., into equivalent electrical signals. The electrically equivalent signals are then converted by the data acquisition system and are then utilized by the end devices in digital form. The ability of the electronic system to preserve signal accuracy and integrity is the main measure of the quality of the system. The basic components required for the acquisition and conversion of analog signals into equivalent digital form are as follows: Analog Multiplexer and Signal Conditioning, Sample/Hold Amplifier, Analog-to-Digital Converter and Timing or Sequence Logic.

Typically, today's data acquisition systems contain all the elements needed for data acquisition and conversion, except perhaps, for input filtering and signal conditioning prior to analog multiplexing. The analog signals are time multiplexed by the analog multiplier; the multiplexer output signal is then usually applied to a very-linear fast-settling differential amplifier and/or to a fast-settling low aperture sample/hold. The sample/hold is programmed to acquire and hold each multiplexed data sample which is converted into digital form by an A/D converter. The converted sample is then presented at the output of the A/D converter in parallel and serial digital form for further processing by the end devices.

2.2.4.1 System sampling rate

2.2.4.1.1 Error Considerations

The application and ultimate use of the converted data determines the required sampling and conversion rate of the data acquisition and conversion system. System sampling rate is determined, as shown in Figure 2.9, by the highest bandwidth channel, the number of data channels and the number of samples per cycle.

2.2.4.1.2 Aliasing Error

From the Nyquist sampling theorem, a minimum of two samples per cycle of the data bandwidth is required in an ideal sampled data system to reproduce sampled data with no loss of information. Thus, the first consideration for determining system sampling rate is aliasing error, i.e., errors due to information being lost by not taking a sufficient number of samples per cycle of signal frequency. Figure 2.10 illustrates aliasing error caused from an insufficient number of samples per cycle of data bandwidth.

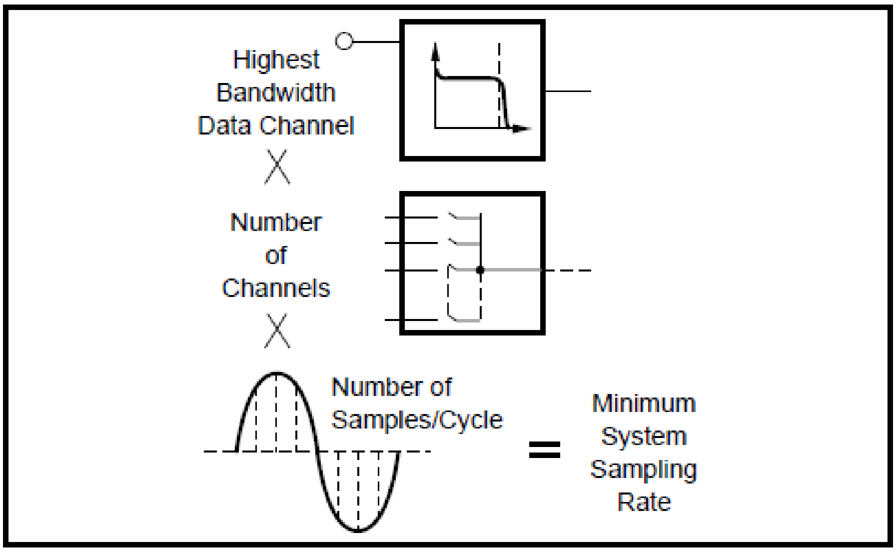


Figure 2.9 Determining minimum system sampling rate.

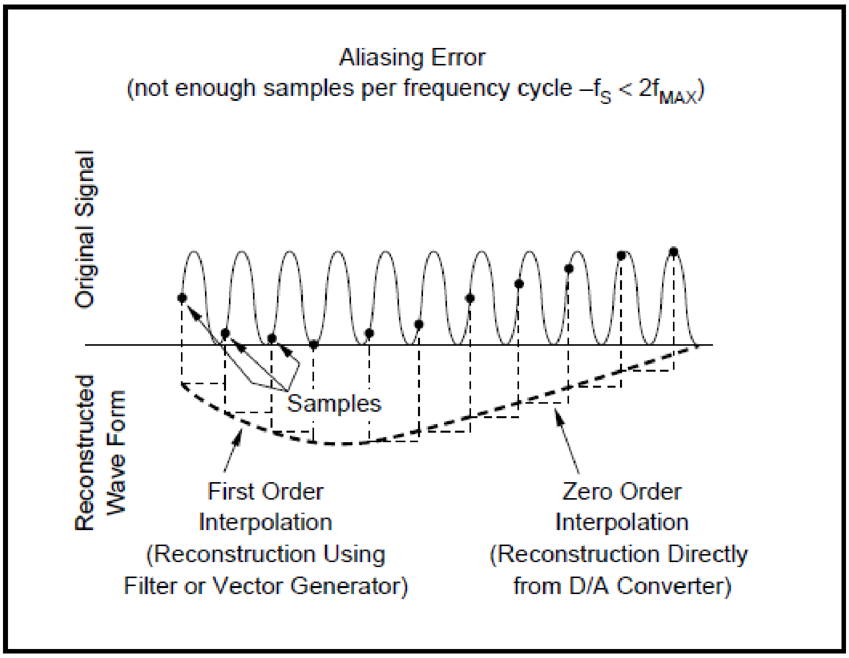


Figure 2.10 Aliasing error versus sampling rate.

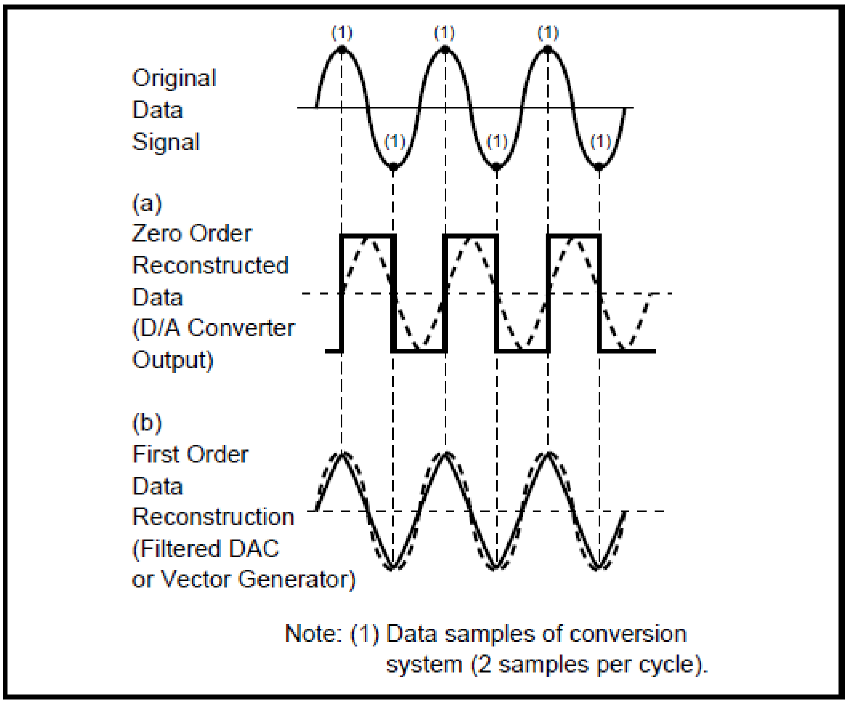


Figure 2.11 Reconstruction of sampled data where $f_S = 2f_{MAX}$.

2.2.4.1.3 Number of Samples per Cycle

This value depends on the allowable average error tolerance, the method of reconstruction (if any), and the end use of the data. Regardless of the end use, the actual error of the discrete data samples will be equal to the throughput error of the data acquisition and conversion system plus any digital errors contributed by a digital computer or other digital end device. For incremental devices such as stepping motors and switches, the average error of sampled digital data is not as important as it is for end devices that require continuous control signals. To illustrate average sampling error in sampled data systems, consider the case where the minimum of 2 samples per cycle of sinusoidal data are taken, and the data is reconstructed directly from an unfiltered D/A converter (zero-order reconstruction). The average error between the reconstructed data and the original signal is one-half the difference in area for one-half cycle divided by p, or 32% for zero order data, and 14% for first order reconstruction. However, the instantaneous accuracy at each sample point is equal to the accuracy of the acquisition and conversion system, and in many applications, this may be sufficient for driving band-limited end devices. The average accuracy of sampled data can be improved by (1) increasing the number of samples per cycle, (2) pre-sample filtering prior to multiplexing, and (3) filtering the D/A converter output.

2.2.4.1.4 Aperture Error

Aperture error is defined as the amplitude and time errors of the sampled data points due to the uncertainty of the dynamic data changes during sampling. In data acquisition and conversion systems, aperture error can be reduced or made insignificant either by the use of a sample/hold or with a very fast A/D converter. For sinusoidal data, maximum aperture error occurs at the zero crossing where the greatest dv/dt occurs, and is expressed mathematically as:

$$\text{Aperture Error} = d\,\frac{(A \sin 2\pi\, ft)}{dt}\ x\ t_A\ x\ 100\%$$

$$= 2\pi\, ft_A\ x\ 100\%\, max \qquad (2.12)$$

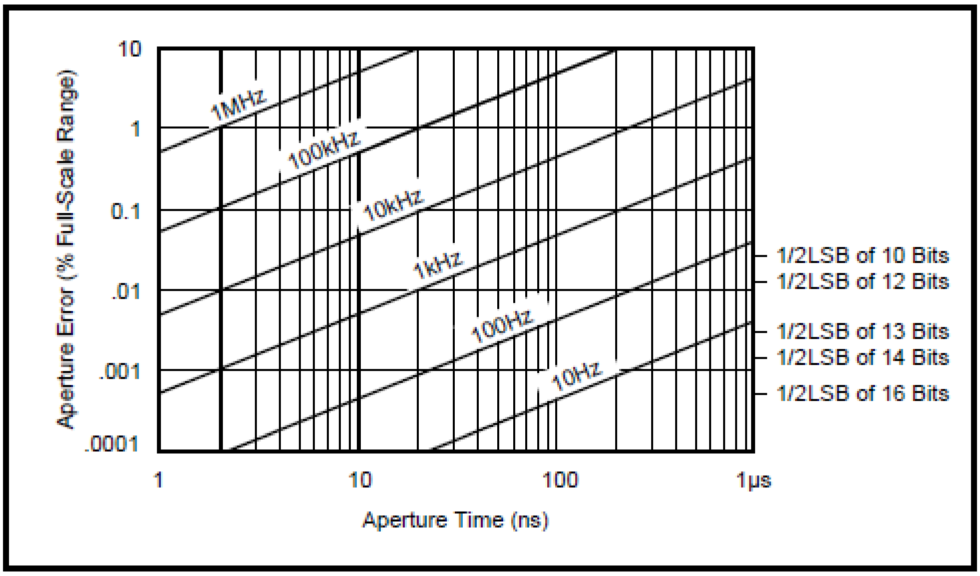


Figure 2.12 Aperture error versus aperture time for data frequencies.

where f is the maximum data frequency and t_A is an aperture time of system. This expression is shown graphically in Figure 2.12 for frequencies of 1Hz to 10kHz with ±1/2LSB error highlighted for 8-, 10- and 12-bit resolution A/D converters. The need for a sample/hold becomes readily apparent when data frequencies of 10Hz or higher are sampled, because the A/D converter conversion speed must be 2ms or faster for aperture errors less than ±l/2LSB for l2-bit resolution, and high speed A/D converters are complicated and expensive when compared to slower A/D converters with a low aperture sample/hold. A sample/ hold with an aperture time of 50ns to 60ns produces negligible aperture error for data frequencies up to 100Hz for 10- and 12-bit resolution A/ D converters, and is less than ±1/2LSB for 8-bit resolution for data frequencies near 5kHz. Use Figure 2.12 to determine your system aperture error for each data channel versus the desired resolution.

2.2.4.2 A/D Converter Points

2.2.4.2.1 Accuracy

All analog values are presumed to exist at the input to the A/D converter. The A/D converter quantizes or encodes specific values of the analog input into equivalent digital codes as an output. These digital codes have an inherent uncertainty or quantization error of ±1/2LSB. That is, the quantized digital code represents an analog voltage that can be anywhere within ±1/2LSB from the mid-point between adjacent digital codes. An A/D converter can never be more accurate than the inherent ±1/2LSB quantizing error. Analog errors such as gain, offset, and linearity errors also affect A/D converter accuracy. Usually, gain and offset errors can be trimmed to zero, but linearity error is unadjustable because it is caused by the fixed-value ladder resistor network and network switch matching. Most quality A/D converters have less than ±1/2LSB linearity error. Another major error consideration is differential linearity error. The size of steps between adjacent transition points in an ideal A/D converter is one LSB. Differential linearity error is the difference between adjacent transition points in an actual A/D converter and an ideal one LSB step. This error must be less than one LSB in order to guarantee that there are no missing codes.

An A/D converter with ±1/2LSB linearity error does not necessarily imply that there are no missing codes.

2.2.4.2.2 Selecting the resolution

The number of bits in the A/D converter determines the resolution of the system. System resolution is determined by the channel(s) having the widest dynamic range and/or the channel(s) that require measurement of the smallest data increment. For example, assume a channel that measures pressure has a dynamic range of 4000psi that must be measured to the nearest pound. This will require an A/D converter with a minimum resolution of 4000 digital codes. A 12-bit A/D converter will provide a resolution of 212 or 4096 codes—adequate for this requirement. The actual resolution of this channel will be 4000/4096 or 0.976 psi. The A/D converter can resolve this measurement to within ±0.488 psi (±1/2LSB).

2.2.4.2.3 Resolution

The number of bits in an A/D converter determines the resolution of the data acquisition system. A/D converter resolution is defined as:

$$\text{Resolution} = \text{One LSB} = \frac{V_{FSR}}{2^n}, \text{ for binary A/D converters}$$

$$= \frac{V_{FSR}}{10^D}, \text{ for decimal A/D converters} \tag{2.13}$$

where LSB is Least Significant Bit, VFSR is Full Scale Input Voltage Range, n is number of bits and D is numbers of decimal digits. The number of bits defines the number of digital codes and is 2^n discrete digital codes for A/D converters.

2.2.5 Wireless Communications

Wireless communications is a type of data communication that is performed and delivered wirelessly. This is a broad term that incorporates all procedures and forms of connecting and communicating between two or more devices using a wireless signal through wireless communication technologies and devices. Wireless communication generally works through electromagnetic signals that are broadcast by an enabled device within the air, physical environment or atmosphere. The sending device can be a sender or an intermediate device with the ability to propagate wireless signals. The communication between two devices occurs when the destination or receiving intermediate device captures these signals, creating a wireless communication bridge between the sender and receiver device.

2.2.5.1 Evolution of LR-WPAN Standardization

Wireless communications is a type of data communication that is performed and delivered wirelessly. This is a broad term that incorporates all procedures and forms of connecting and communicating between two or more devices using a wireless signal through wireless communication technologies and devices. Wireless communication generally works through electromagnetic signals that are broadcast by an enabled device within the air, physical environment or atmosphere. The sending device can be a sender or an intermediate device with the ability to propagate wireless signals. The communication between two devices occurs when the destination or receiving intermediate device captures these signals, creating a wireless communication bridge between the sender and receiver device.

During the mid-1980s, it turned out that an even smaller coverage area is needed for higher user densities and the emergent data traffic. The IEEE 802.11 working group for WLANs is formed to create a wireless local area network standard.

Whereas IEEE 802.11 was concerned with features such as Ethernet matching speed, long range (100m), complexity to handle seamless roaming, message forwarding, and data throughput of 2-11Mbps, WPANs are focused on a space around a person or object that typically extends up to 10m in all directions. The focus of WPANs is low-cost, low power, short range and very small size. The IEEE 802.15 working group is formed to create WPAN standard. This group has currently defined three classes of WPANs that are differentiated by data rate, battery drain and quality of service (QoS). The high data rate WPAN (IEEE 802.15.3) is suitable for multi-media applications that require very high QoS. Medium rate WPANs (IEEE 802.15.1/Blueetooth) will handle a variety of tasks ranging from cell phones to PDA communications and have QoS suitable for voice communications. The low rate WPANs (IEEE 802.15.4/LR-WPAN) is intended to serve a set of industrial, residential and medical applications with very low power consumption and cost requirement not considered by the above WPANs and with relaxed needs for data rate and QoS. The low data rate enables the LR-WPAN to consume very little power.

2.2.5.2 ZigBee and IEEE 802.15.4

ZigBee technology is a low data rate, low power consumption, low cost, wireless networking protocol targeted towards automation and remote control applications. IEEE 802.15.4 committee started working on a low data rate standard a short while later. Then the ZigBee Alliance and the IEEE decided to join forces and ZigBee is the commercial name for this technology. ZigBee is expected to provide low cost and low power connectivity for equipment that needs battery life as long as several months to several years but does not require data transfer rates as high as those enabled by Bluetooth. In addition, ZigBee can be implemented in mesh networks larger than is possible with Bluetooth. ZigBee compliant wireless devices are expected to transmit 10-75 meters, depending on the RF environment and the power output consumption required for a given application, and will operate in the unlicensed RF worldwide (2.4GHz global, 915MHz Americas or 868 MHz Europe). The data rate is 250kbps at 2.4GHz, 40kbps at 915MHz and 20kbps at 868MHz.

IEEE and ZigBee Alliance have been working closely to specify the entire protocol stack. IEEE 802.15.4 focuses on the specification of the lower two layers of the protocol (physical and data link layer). On the other hand, ZigBee Alliance aims to provide the upper layers of the protocol stack (from network to the application layer) for interoperable data networking, security services and a range of wireless home and building control solutions, provide interoperability compliance testing, marketing of the standard, advanced engineering for the evolution of the standard. This will assure consumers to buy products from different manufacturers with confidence that the products will work together. IEEE 802.15.4 is now detailing the specification of PHY and MAC by offering building blocks for different types of networking known as “star, mesh, and cluster tree”. Network routing schemes are designed to ensure power conservation, and low latency through guaranteed time slots. A unique feature of ZigBee network layer is communication redundancy eliminating “single point of failure” in mesh networks. Key features of PHY include energy and link quality detection, clear channel assessment for improved coexistence with other wireless networks.

2.2.5.3 ZigBee vs. Bluetooth

ZigBee looks rather like Bluetooth but is simpler, has a lower data rate and spends most of its time snoozing. This characteristic means that a node on a ZigBee network should be able to run for six months to two years on just two AA batteries. The operational range of ZigBee is more than 100m compared to 10m for Bluetooth (without a power amplifier). ZigBee sits below Bluetooth in terms of data rate. The data rate of ZigBee is 250kbps at 2.4GHz, 40kbps at 915MHz and 20kbps at 868MHz whereas that of Bluetooth is 1Mbps. ZigBee uses a basic master-slave configuration suited to static star networks of many infrequently used devices that talk via small data packets. Bluetooth’s protocol is more complex since it is geared towards handling voice, images and file transfers in ad hoc networks.

Bluetooth devices can support scatternets of multiple smaller non-synchronized networks (piconets). It only allows up to 8 slave nodes in a basic master-slave piconet set-up. When ZigBee node is powered down, it can wake up and get a packet in around 15 msec whereas a Bluetooth device would take around 3sec to wake up and respond.

A ZigBee system consists of several components. The most basic is the device. A device can be a full-function device (FFD) or reduced-function device (RFD). A network shall include at least one FFD, operating as the PAN coordinator. The FFD can operate in three modes: a personal area network (PAN) coordinator, a coordinator or a device. An RFD is intended for applications that are extremely simple and do not need to send large amounts of data. An FFD can talk to RFDs or

FFDs while an RFD can only talk to an FFD. The comparison of ZigBee, Bluetooth and WiFi is shown as Table 2.1.

Table 2.1 Comparison of wireless standards.

No.	Item	Type		
		ZigBee	**WiFi**	**Bluetooth**
1	Standard	IEEE 802.15.4	IEEE 802.11a, b, g	IEEE 802.15.1
2	Application	Monitoring, control	Web, e-mail, video	Cable replacement
3	System resources	50 to 60 kbytes	> 1 Mbyte	> 250 kbytes
4	Battery life (days)	100 to > 1,000	1 to 5	1 to 7
5	Network size	65,536	32	7
6	Bandwidth (kbps)	20 to 250	11,000	720
7	Maximum transmission range (m)	> 100	100	10
8	Success metrics	Reliability, power, cost	Speed, flexibility	Cost, convenience

2.2.5.4 Network Topologies

Figure 2.13 shows 3 types of topologies that ZigBee supports: star topology, peer-to-peer topology and cluster tree.

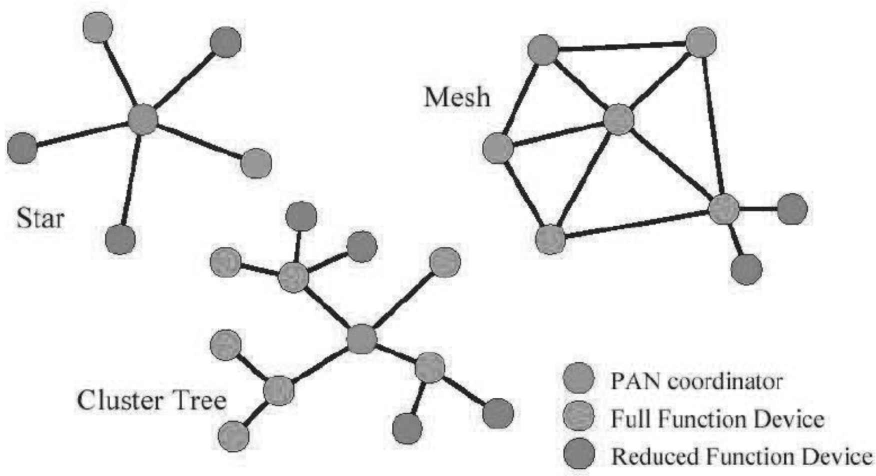


Figure 2.13 Topology Models.

2.2.5.4.1 Star Topology

In the star topology, the communication is established between devices and a single central controller, called the PAN coordinator. The PAN coordinator may be mains powered while the devices will most likely be battery powered. Applications that benefit from this topology include home automation, personal computer (PC) peripherals, toys and games. After an FFD is activated for the first time, it may establish its own network and become the PAN coordinator. Each start network chooses a PAN identifier, which is not currently used by any other network within the radio sphere of influence. This allows each star network to operate independently.

2.2.5.4.2 Peer-to-peer Topology

In peer-to-peer topology, there is also one PAN coordinator. In contrast to star topology, any device can communicate with any other device as long as they are in range of one another. A peer-to-peer network can be ad hoc, self-organizing and self-healing. Applications such as industrial control and monitoring, wireless sensor networks, asset and inventory tracking would benefit from such a topology. It also allows multiple hops to route messages from any device to any other device in the network. It can provide reliability by multipath routing.

2.2.5.4.3 Cluster-tree Topology

Cluster-tree network is a special case of a peer-to-peer network in which most devices are FFDs and an RFD may connect to a cluster-tree network as a leave node at the end of a branch. Any of the FFD can act as a coordinator and provide synchronization services to other devices and coordinators. Only one of these coordinators however is the PAN coordinator. The PAN coordinator forms the first cluster by establishing itself as the cluster head (CLH) with a cluster identifier (CID) of zero, choosing an unused PAN identifier, and broadcasting beacon frames to neighboring devices. A candidate device receiving a beacon frame may request to join the network at the CLH. If the PAN coordinator permits the device to join, it will add this new device as a child device in its neighbor list. The newly joined device will add the CLH as its parent in its neighbor list and begin transmitting periodic beacons such that other candidate devices may then join the network at that device. Once application or network requirements are met, the PAN coordinator may instruct a device to become the CLH of a new cluster adjacent to the first one. The advantage of this clustered structure is the increased coverage area at the cost of increased message latency.

2.2.5.5 LR-WPAN Device Architecture

Figure 2.14 shows an LR-WPAN device. The device comprises a PHY, which contains the radio frequency (RF) transceiver along with its low-level control mechanism, and a MAC sublayer that provides access to the physical channel for all types of transfer. The upper layers consists of a network layer, which provides network configuration, manipulation, and message routing, and application layer, which provides the intended function of a device. An IEEE 802.2 logical link control (LLC) can access the MAC sublayer through the service specific convergence sublayer (SSCS).

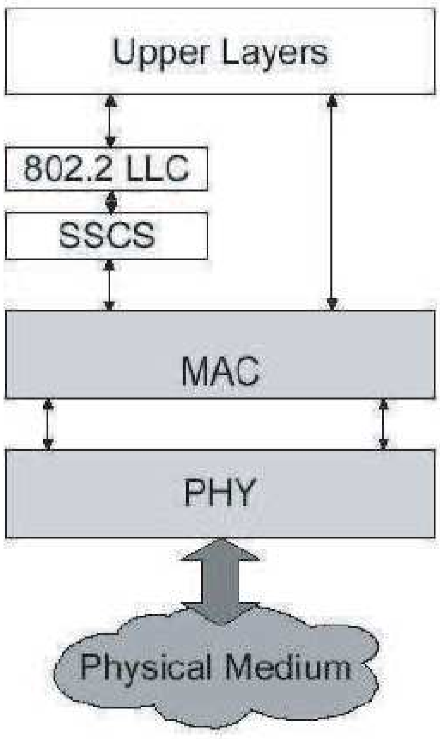


Figure 2.14 LR-WPAN Device Architecture.

2.3 Literature Reviews

AE sensors are applied in many applications which various method and techniques are utilized. Table 2.2 summarizes research on the applications, techniques and characteristics of AE sensor for pipelines fault detection as follows.

Table 2.2 Summary applications and techniques of AE sensors.

Author	Year	Application	Theory/Method	AE Sensor
Jiao Yang	2006	AE Source Identification for Buried Gas Pipeline Leak	Global Spatial-Temporal Data Fusion	-
Athanasios A.	2009	Leak Detection in Liquid Filled Buried Pipeline	Reynol Number of Turbelent Flow	-
M. Noipitak	2011	Fabricated AE Sensor Caibration by the leakage rate of Air Jet	Empirical Regression Theoretical Model	150 kHz
A. Prateepasen	2011	Apply Smart Portable AE Instrument to detect internal air leakage of valve	Use Microcontroller ARM7 to control the multiplexer circuit to get Aerms Mathematical modelling	150 kHz
Zhang Jain-li	2011	Experiment about Leakage Detection in Cast Iron Pipe	Detect AE Signal that generated from Turbulent Flow	0.5 - 10 kHz
Diem Ozevin	2012	Leak Localization in pipeline networks	Geometric Connectivity	60 kHz
N.C. Hii	2013	Flow of particulate solids detection in pipelines	Detect flow rate of solids partical by AE sonsors	0.1 - 1 MHz
A. Mostafapour	2013	Leakage Detection in Pipeline	Donnell's non-linear cylindrical shell Galerkin Method Runge-Kutta numerical method	50 - 500 kHz
Eric Bechhoefer	2013	Split Torque Gearbox Abnormal Detection	Analog Signal Processing for the AE Signature Demodulating	100 - 900 kHz
Zhang Haifeng	2013	Valve Leak Detection in Pipeline	Support Vector Machine RBF Kernel Function	-

As shown in Table 2.2, Jiao Yang; et al. (2006) [5] have proposed a gas leak identification model for city gas pipeline. This model is presented based on spatial-temporal data fusion, which the leak in small holes (diameter 1 – 2 mm) can be detected with the distance between leak hole and testing sensor is within 87 m, and the pressure in pipeline is in the range of 0.2 - 1.0 MPa. The identification ratio is up to 95%.

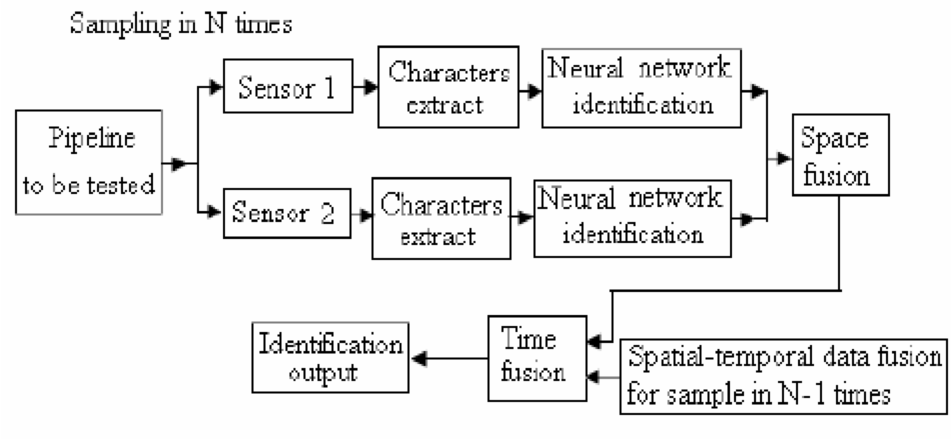


Figure 2.15 Spatial-temporal data fusion identification model for leak testing.

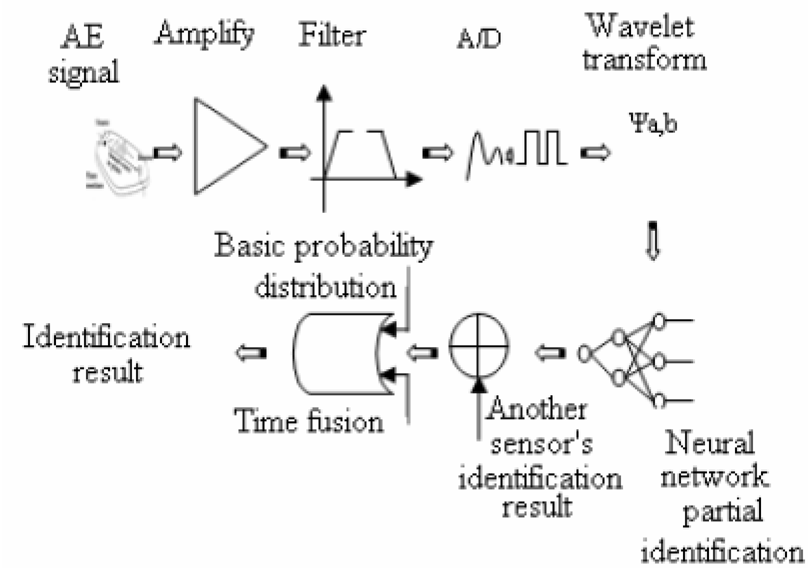


Figure 2.16 The realizing process of leak identification.

Figure 2.16 shows the process of AE signal identification. In the course of leak AE testing and identification, several seconds before acquisition time, GPS clock wakes single-piece microcomputer up. The computer starts to calibrate precisely time and labels GPS time tag. When acquisition time is up, the instrument informs A/D to acquire signal. When there is leak in inspected pipeline, the AE signal spreads in guided wave model and received by two sensors installed one on each end of pipeline. After amplifying and filtering, the analog signals are converted to digital signal by 500 kHz A/D converter, and the latter are stored in FLASH of the acquisition instrument via buffer. The acquisition instrument acquires signals according to acquisition times and length which were set previously. The whole data are stored in FLASH and provided for host computer to read and analysis. Host computer reads data via USB port. There is the precise GPS time tag in every segment data, which is for locating leak source. The author employed wavelet analysis method to pre-process original signal features because it has better time-frequency localization characteristic for stochastic signal. The acquisition data treated with wavelet to decreased noise are identified locally by neural network with distributed decision, and finally identified with global spatial-temporal data fusion in host computer. The extracting features are sent to neutral network to make locally target classification of a single sensor. The output value is used to make $m_i(A)$ and $m_i(\theta)$ functions, and then the final basic credibility distribution function is acquired by the Dempster amalgamate rule. The identification result is at last fused in new temporal data series. In accordance with D-S evidence rule, the final result about target identification is acquired.

Athanasios Anastasopouos; et al. (2009) [6] have applied AE sensors to detect the turbulent flow at the leak orifice. The author have provided the position of the leak using digital AE systems and specialized software. In this research, the author mentioned that AE can be produced by the highly unstable turbulent pressure field at the orifice with the Reynolds number Re > 1000, so as to ensure turbulent flow. The corresponding AE signals generated are of a “continuous” nature. Additional sources that may produce AE in the occasion of a leak are local crack/orifice growth, cavitation due to local sub-pressure at the orifice, temporary entrapments and impacts of solid particles at the orifice, soil movements, or even external sources such as impacts etc., which are mainly “burst” type sources. The generated AE waves from such sources propagate through the fluid or through the pipeline itself. The author estimated the leak location by measuring the amplitude variations of continuous signal at various positions along the pipe. Based on signal attenuation and signal amplitude reduction with the distance from the leak source, as measured at various positions, an amplitude variation ratio is recorded. Based on this ratio the distance to the source can be roughly calculated. They also presented a more effective and accurate method to locate a leak on a buried pipeline by linear location. Two AE sensors placed on either side of the leak are required for this method. The leak source can be calculated as equation 2.14.

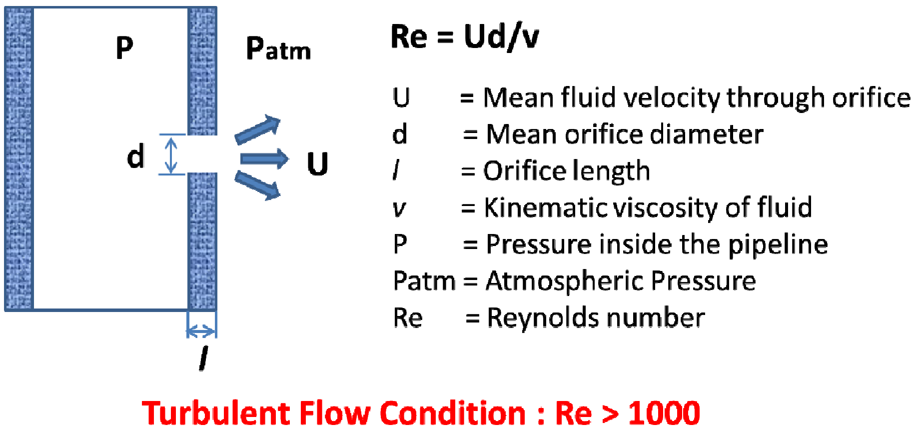


Figure 2.17 Leaking flow features.

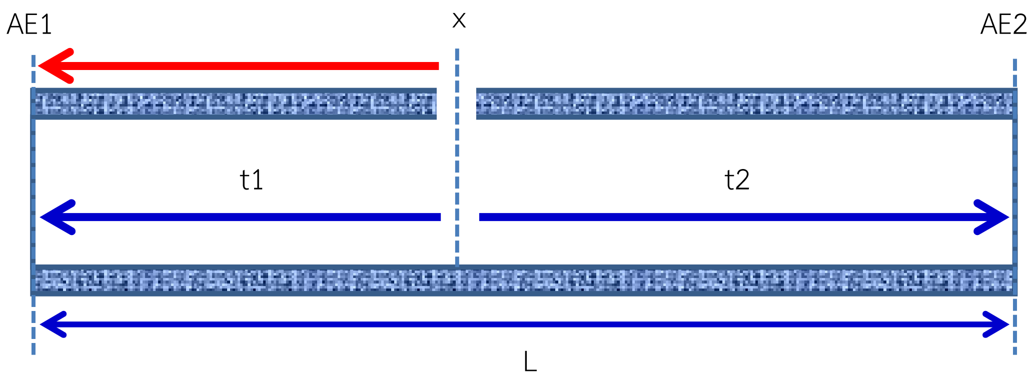


Figure 2.18 AE sensors installation for leak source identification.

$$x = \frac{(L - V\nabla t)}{2} \tag{2.14}$$

Where x is the distance from the first sensor, L is the known distance between the two sensors, V is the (known or measured) AE wave velocity, and ∇t is the time difference of the wave arrival on the two sensors measured by the acquisition system.

M. Noipitak; et al. (2011) [7] proposed a relative calibration method for an internal valve leakage rate measurement system using acoustic emission (AE) methods with microcontroller. The author selected an air jet to calibrate because it provides a frequency spectrum similar to the AE spectrum obtained from the valve leak, especially in the frequency range of 100– 300 kHz. Three AE sensors mounted on a valve were used to validate the system calibration. Ratios between the average energy (AE_{RMS}) were obtained from each pair of sensors, and these ratios were used to transfer the leakage AE_{RMS} value. Subsequence, its performance to predict the leakage rate in the laboratory and in the field was tested. The error is less than 5% in almost cases. Its benefit is to reduce the recalibration time when a part of measurement system is changed. Accordingly, inexpensive equipment including an AE sensor was built, and its system performance was revealed. The relationship between leakage rate and average energy of AE signal (AE_{RMS}) is used to predict leakage rate and was determined both by an empirical regression and a theoretical model. The equation is based on a reference AE sensor, which has a frequency response that can differ significantly compared to other AE sensors. As a consequence, changing an AE sensor can produce a large prediction error when using these leakage rate equations. Therefore a time-consuming calibration process must be performed to acquire the correct parameters for the leakage rate equations. To minimize the time required for calibration, a relative calibration method using an air jet is introduced here. In this paper, a system calibration will be applied to a valve leakage rate measurement system, using an air jet as the artificial AE source. An AE sensor is built for the valve leakage rate measurement system. A method for transferring information between different sensors is also proposed. The prediction error, in both the laboratory and a petrochemical plant, is presented. The contribution of this paper is the development of a novel, low-cost valve leakage rate measurement system, which can minimize the time and cost requirements for making a useful leakage rate prediction.

A. Prateepasen; et al. (2010) [8] proposed a novel low-cost instrument based on microcontroller and a novel theoretical model based on AE technique to predict the leakage rate. The system is an embedded system instead of a general PC-based data acquisition. AE_{RMS} parameter is used to infer the leakage rate, and the effects of various process variables on the model are also studied. The experimental results have shown that the instrument is capable of detecting possible valve leakage encountered in online operation. With its portability, ease of use and compactness, the proposed system provides faster and low cost valve leakage detection. The author mentioned about a theoretical relationship between an AE signal power in form of AE_{RMS}^2 to a liquid leakage rate acquired from a PC-based AE system. The AE_{RMS}^2, generated from the leakage of gas through valves, is a function of sound power, P_s, related to fluid variables such as inlet pressure level, valve size, gas density and sound velocity. This can be seen from the ratios (δ) of AE_{RMS}^2, to P_s at various air leakage rate conditions of valve as shown in Figure 2.19. The gas compressibility under the turbulent flow at thin layer with a rigid boundary affect sound generation of nonlinear deformations of material surfaces. The expression between the leakage rate, AE_{RMS}^2 and the fluid variables can be expressed as

$$Q = f(\beta), \tag{2.15}$$

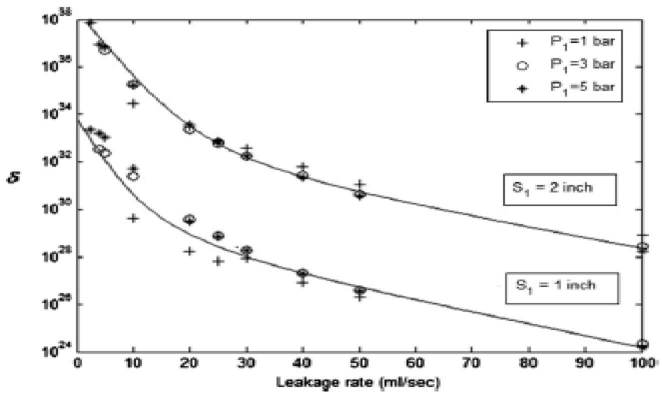


Figure 2.19 Correlation between d and leakage rate (Q) at various parameters.

where $\beta = \left[\frac{AE_{RMS}^{2}\alpha^{5} D^{14} RT}{P_1}\right]^{\frac{1}{8}}$ and Q is the volume flow rate (m^3/s), AE_{RMS}^{2} the AE signal power (mV^2) obtained by experiment, α the sound velocity in the fluid (m/s), D the valve size (m), R the gas constant ($N\ m\ kg^{-1}\ K^{-1}$), T the temperature (K) and P_1 the inlet pressure (kg/m S^2).

Zhang Jian-li; et al. (2011) [9] took the cast iron pipe as the research object, several cases of acoustic emission phenomenon in the leaking pipe were studied by experiment in the municipal pipeline test bench. They are the different characteristic of the signals from the pipe with leakage and no-leakage, the different acoustic emission characteristics of the leakage in cast iron pipe under different hydraulic conditions, and some of the external interference to the acquisition of signals, etc. The experiment results show that the acoustic emission phenomenon of leakage in the pipe is very complex and the components of the frequencies in the leak noises are very rich. The autocorrelation of leak signals is poor; however the autocorrelation of the ambient noises is very strong. And the impact of different hydraulic conditions on amplitude is evident but the distribution of the spectrum is not obvious.

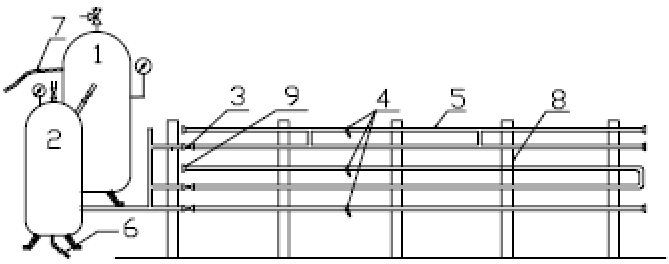


Figure 2.20 Test bench for leakage simulation.

In this paper, the characteristics of leak signals in cast iron water distribution pipe have been investigated though time-domain, frequency-domain, time-frequency domain, and the autocorrelogram under many controlled conditions at an experimental leak detection bench. The main findings of the investigation can be summarized as 1) The energy of leak noise is much higher than the ambient noises. Furthermore, the greater the leak flow rate the higher the energy is, 2) The amplitude of leak signals diminished rapidly with distance, at a rate of about 0.11dB/m, 3) The spectrums of leak signals distribute widely. The spectrums induced by water flow are in low frequency that below 3000Hz and the spectrums induced by bubbles burst are in high frequency that above 5000Hz, 4) The autocorrelogram of leak noise is sharper than that of ambient noise much more.

Didem Ozevin; et al. (2012) [10] proposed a new leak localization approach for pipeline networks spread in a two dimensional configuration. The approach is to determine arrival time differences using cross correlation function, and introduce the geometric connectivity in order to identify the path that the leak waves should propagate to reach the AE sensors. The leak location in multi-dimensional space is identified in an effective approach using an array of sensors spread on the pipeline network. The approach is successfully demonstrated on laboratory scale polypropylene pipeline networks. This paper demonstrates that the 2D location of a leak in a pipeline network can be determined using the 1D source location algorithm integrated with geometric connectivity. As leak source generates continuous acoustic emissions which can be highly chaotic, cross correlation approach should be integrated with the hit sequence identification based on the ASL distributions of the AE sensors in order to reduce the error of the arrival time differences of the AE sensors. The source location algorithm presented in this study is applicable to any novel AE sensors such as fiber optic and MEMS. The attenuation and the wave velocity study with distance can be integrated with the location model for any kinds of pipeline materials in order to increase the reliable leakage location.

N.C. Hii; et al. (2013) [11] researched about the generation of the AE from particulate flow and an investigation of the potential of implementing AE for flow parameters, namely the solid mass flow rate, particle velocity and size, monitoring. A series of experiments has been conducted to gather AE signals from a laboratory scale single flow-loop pneumatic conveying system. Initially, AE sensors were attached to two steel meshes which were placed with a fixed axial distance in the pipeline to study the generation of the AE and subsequently the possibility of using those generated AE to determine particle velocity in the pipeline. Particle velocities measured from this approach were compared with theoretical predictions. The results indicated that this approach could measure the mean particle velocity with reasonable accuracy. The generation of AE on five different sensor mounting locations was also studied. The results showed that sensors mounted on all those locations were able to respond to changes in the flow parameters. However, only two sensor locations (outer bend and Mesh) were chosen for further investigation. The final experimental results indicated that the AE features, namely Root-Mean-Square (RMS) and energy of the AE, are related to the changes in the flow parameters and good correlations were found. Good correlations between the RMS and energy of the AE with the momentum and kinetic energy of the particles, respectively, were also found. Overall, the studies indicated that features of AE have great potential in gas–solid two phase flow parameter monitoring. However, the studies also show that the applicability of the AE techniques to measure solid mass flow rates in practice would require tedious calibration.

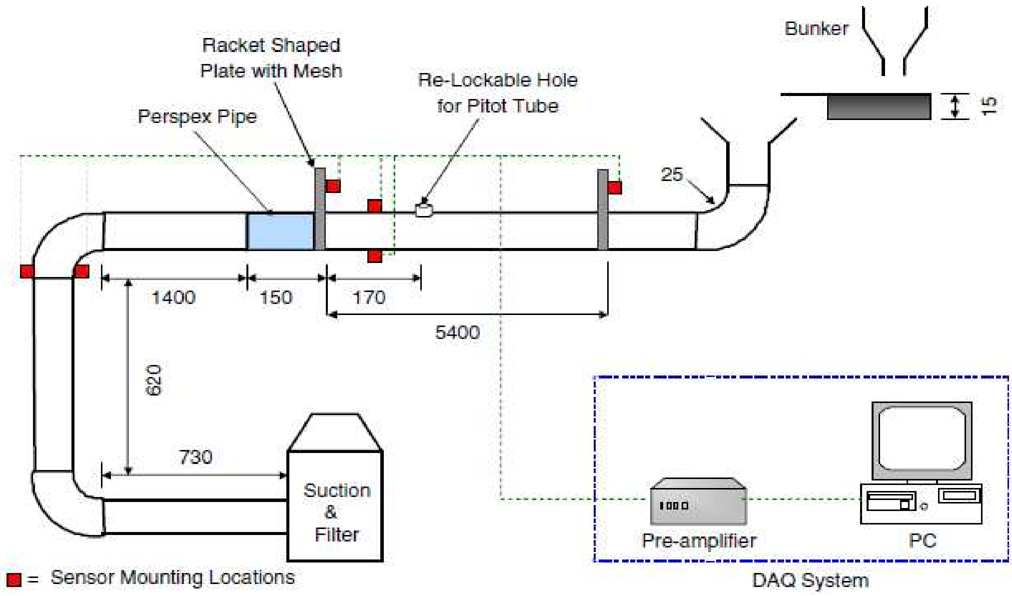


Figure 2.21 Schematic of the laboratory scale single flow-loop pneumatic conveying system (all dimensions in mm).

Their experiments cover briefly three main investigations, which are 1) The effects of particle size and solid loading on the mean particle velocity, 2) The effect of different sensor location on AE generation, 3) The effects of changes in the flow parameters, namely solid mass flow rate and conveying air velocity on the AE features. The author concluded that the flow of particulate solids in pipeline results in the generation of AE which may be detected by sensors installed at several different locations along the pipe outer surface. It was found that the particle slip velocity decreases linearly with increasing solid loading for a given particle size and conveying air velocity. Moreover the energy of the AE signals has been shown to correlate linearly, with reasonable accuracy, with respect to the kinetic energy of the particles. Although the AE sensors responded to changes in the solid mass flow rates in a systematic way, the relationships were generally not linear in nature and that the outcomes were very sensitive to the location and coupling of the sensor installation. Furthermore it is also anticipated the absolute results of AE measurements are dependent on the pipe material and diameter and the physical properties of particulate solid being conveyed. This means that the applicability of the AE techniques to measure solid mass flow rates in practice would be subject to tedious calibration.

A.Mostafapour; et al. (2013) [12] proposed a model of acoustic emission generated by pipe vibration due to leakage. Donnell's non-linear theory for cylindrical shell was used to derive motion equation under simply supported boundary condition. Then, the motion equation was solved by using Galerkin method that resulted in a system of non-linear equations with 6 degrees of freedom. To solve these non-linear equations, ODE tool of MATLAB software and Runge–Kutta numerical method was employed to obtain pipe wall radial displacement. For verifying this method, acoustic emission by a continuous leak source was measured. Experiments were carried out with a linear array of sensors on steel pipe (ASTM A 106/99) of nominal length 6 m, 7.35 mm wall thickness and external diameter of 169 mm. The pressurized air was flown inside the pipe through the compressor. Two simulated continues leak sources with 0.6-mm and 1-mm diameter holes were used under 5 bar air pressure. This source propagated waves in a large of frequencies about 0–400 kHz.

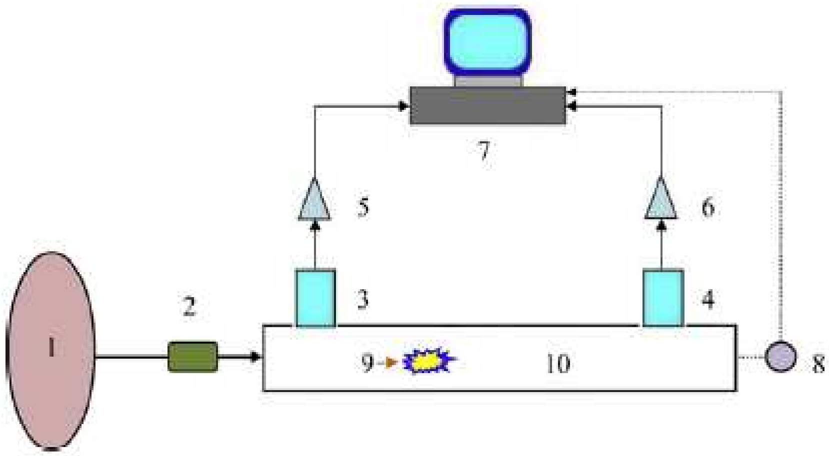


Figure 2.22 Equipment lay-out (1 – air compressor; 2 – regulator; 3, 4 – AE sensors; 5, 6 – amplifiers; 7 – processing PC; 8 – pressure gauge; 9 – leak source and 10 – pipe).

In this study the vibration behavior of the pipe is investigated per resonance frequencies of the used AE sensors which are near 150 and 300 kHz. Signals generated by the pipe wall vibration were recorded by using acoustic emission sensors. In the next step, Fast Fourier Transform (FFT) was used in the signal analysis. Comparison of the obtained results, indicate the good agreement between the experimental and modeled frequencies ranges. The mean error between analytical modeling and experimental results is less than 6%. So the experimental results confirm performance of the developed analytical.

Eric Bechhoefer; et al. (2013) [13] hypothesized that the AE signature is the result of some forcing function (e.g. periodic motion in the case of rotating machinery). By using analog signal processing to demodulating the AE signature, one can reconstruct the information carried (e.g. modulation) by the AE signature and provide improved diagnostics. As most on-line condition monitoring systems are embedded system, analog signal processing techniques where used which reduce the effective sample rate needed to operate on the AE signal to those similarly found in traditional vibration processing systems. Further, by implementing another signal processing technique, time synchronous averaging, the AE signal is further enhanced. This allowed, for the first time, an AE signal to be used to identify a specific component within gearbox. This processing is tested on a split torque gearbox and results are presented. The AE envelope analysis show promises to be a powerful tool for gear fault diagnostics. By heterodyning the raw AE signal, it is possible to reduce the hardware resources and cost normally associated with AE processing. In this experiment, the acquisition-sampling rate of 100 KSPS was used on an AE sensor with a signal bandwidth of 600 KHz, using an analogy Hilbert transform circuit. The AE envelope signal was then processed using time synchronous averaging (TSA). The TSA is commonly used with vibration-based diagnostics: this is the first time its use has been published using AE data. The TSA of the AE envelope was used to control for variation in shaft speed, and to reduce non-synchronous noise. The use of the TSA allowed the gear fault to be identified.

Condition Indicators, based on the TSA, were calculated for both the AE sensor and the for vibration sensor (accelerometer). The CIs for the AE enveloped signal were 3x more statistically significant than for the vibration sensor. This indicated that the combination of demodulated AE sensor data and the use of the TSA was superior for gear fault detection than traditional vibration/accelerometer sensors.

Zhang Haifeng; et al. (2013) [14] proposed a method to detect the leakage of the pipeline valve in operation sate based on acoustic emission (AE) theory and support vector machine (SVM) model. The acoustic emission testing platform was setup, and then, AE testing for valve internal leakage under test platform was performed, and the root mean square (RMS), average signal level (ASL) of the time domination and peak value of the frequency domination were as eigenvectors for the SVM model. Finally, the SVM model for the detection of leakage of pipe valve was established through the training and testing eigenvectors, and the abilities of the kernel functions were evaluated. Results show that the method based on RBF kernel function is workable and effective for the leak detection of pipe valve with the sensitivity of 92.5%, the specificity of 100%, and the accuracy of 96.25%.

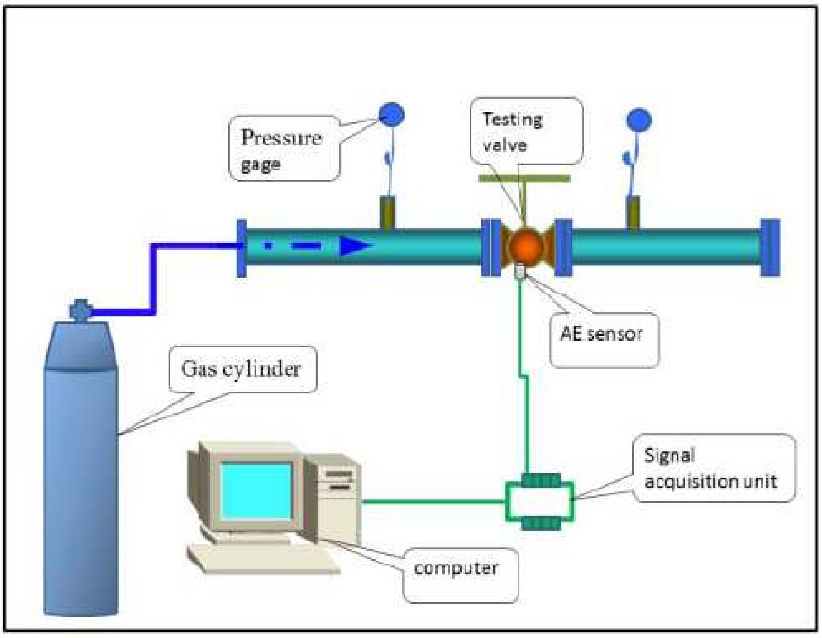


Figure 2.23 Valve leakage detection test platform.

Chapter 3
Research Methodology

3.1 Introduction

The research methodology of this thesis focuses on a real-time monitoring of closed-loop water cooling system using wireless AE signal transmission combined with a microcontroller–based data acquisition system and a Graphic User Interface (GUI) in MATLAB. The application of AE signal in turbulent flow detection in the pipeline is also focused in this research. This chapter presents the research methodology for implementation process.

3.2 Research Processes

3.2.1 Review related research papers and studies from IEEE explorer and Science Direct databases not only to ensure that this research is not a repeat, but also to study the existing knowledge and technology that can be applied for this thesis.

3.2.2 Design and implement low-pass filter and pre-amplifier circuits using BiMOS operational amplifier model CA3140, which the non-inverting amplifier concept is applied for this design.

3.2.3 Design and implement an A/D converter by programming on STM32F4 microcontroller. This A/D converter converts the AE signals to a personal computer and hence the GUI software. Simulations are performed through Orcad PSpice and a MATLAB Simulink. Signal quality analysis involves frequency response, Total Harmonic Distortion (THD), and Power Spectral Density (PSD).

3.2.4 Design and program on XBee wireless module to transmit wirelessly at the frequency of 2.4 GHz that receives digital signals from preceding A/D converter and sends data to another wireless module connected to a personal computer via USB port.

3.2.5 Design and implement GUI with data analysis software by programming on MATLAB to display AE signals on a personal computer in both time and frequency domains.

3.2.6 Test the overall system with the closed-loop water cooling system of the induction heater in the galvanizing steel plant to find the relationship between AE signals in both time and frequency domains at particular conditions of water pressures and flow rates.

3.3 Research Tools

This research acquires the AE signals that generated from AE sensor model R80α of Physical Acoustic Corporation (PAC) which has the operating frequency up to 1 MHz, with the pre-amplifier in which the gains can be selected at 20, 40 and 60dB. The commercially available DAQ board used as for a reference to the experimental result is from ADVANTECH model PCIE-1744, having the maximum sampling rate at 30 MHz with 12 bits resolution.

3.4 Data Analysis Methods

The signal analysis of the proposed system is performed through MATLAB. Signal quality analysis involves frequency response, Total Harmonic Distortion (THD), and Power Spectral Density (PSD). The results are also compared with the real commercially available DAQ system with LabVIEW analysis in order to ensure the reliability of the proposed system. The commercially available AE sensor and DAQ system is calibrated using the standard pencil lead break tests on an aluminum plate according to the artificial source, i.e. ASTM Standard No. E976-84 [15].

Chapter 4
Research Results

4.1 Introduction

This chapter presents the research results for a Real-Time Monitoring of a Closed-Loop Water Cooling System using a Wireless Acoustic Emission Sensor. The proposed system consists of three major blocks as shown in Figure 4.1. First, an integrated amplifier and filter, which is so called preamplifier, that is used to amplify the input signal with variable gain and filter noises in environment. Second, the A/D converter interface using STM32F4 controller which operates real-time. Afterward, XBee wireless module is programmed to transmit AE signal wirelessly to another wireless module connected to a personal computer via USB port. Subsequently, the graphic user interface system exploits MATLAB display capability to show the real-time signal and various signal characteristics analysis. Finally, overall system is tested with the closed-loop water cooling system of the induction heater in the galvanizing steel plant to find the relationship between AE signals in both time and frequency domains. This chapter summarizes the resulting works of this thesis involving all theoretical matters and experimental verifications.

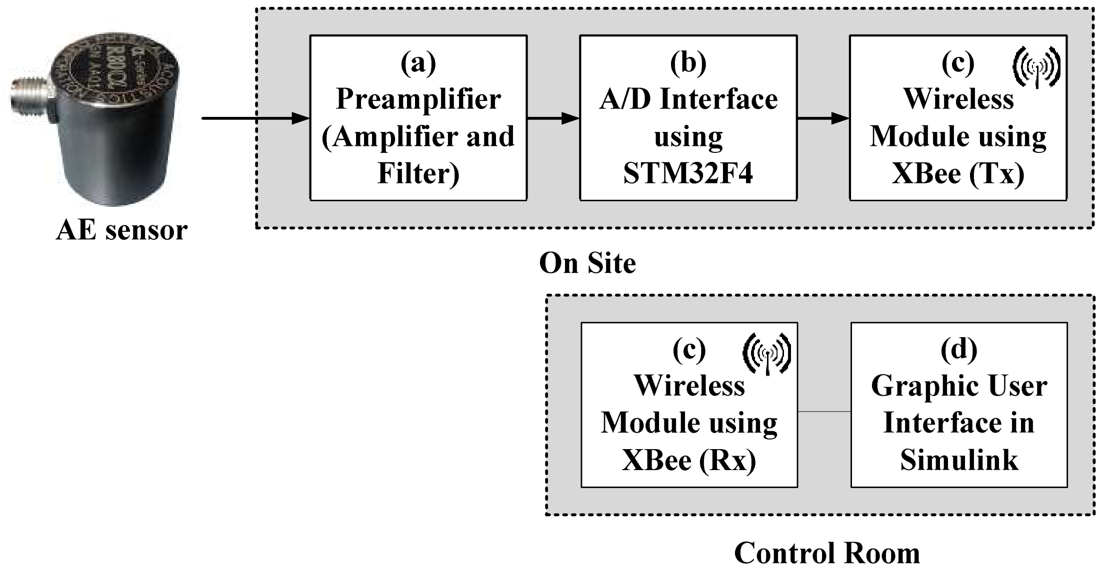


Figure 4.1 The proposed overall research scope.

4.2 The Design and Implementation of Pre-amplifier circuits

Figures 4.2 and 4.3 show the preamplifier circuit designs. The variable gain preamplifier is implemented using BiMOS operational amplifier model CA3140 as a high slew rate, wideband amplifier. The amplifier circuit is designed base on a non-inverting amplifier which analog input signal is connected at the input terminal which signal level can be adjusted by R1 before input of CA3140. Gain of preamplifier can be adjusted via R9 and output DC level can also be adjusted by R5 to keep output signal at the output terminal in the range of 0-5 V that compatible with STM32F4 Controller. The push-pull class AB output circuit using transistors Q1 and Q2 are exploited in order to adjust full peak-to-peak swing.

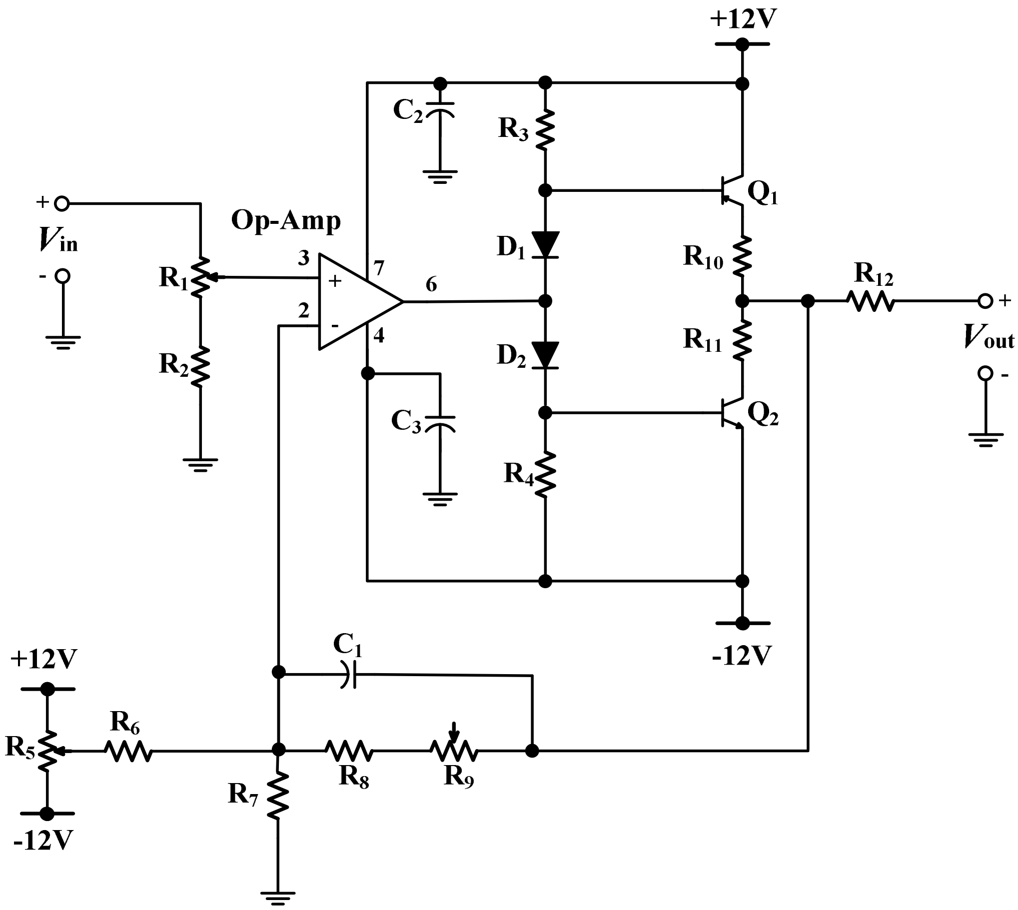


Figure 4.2 The circuit schematic diagram of preamplifier circuit design.

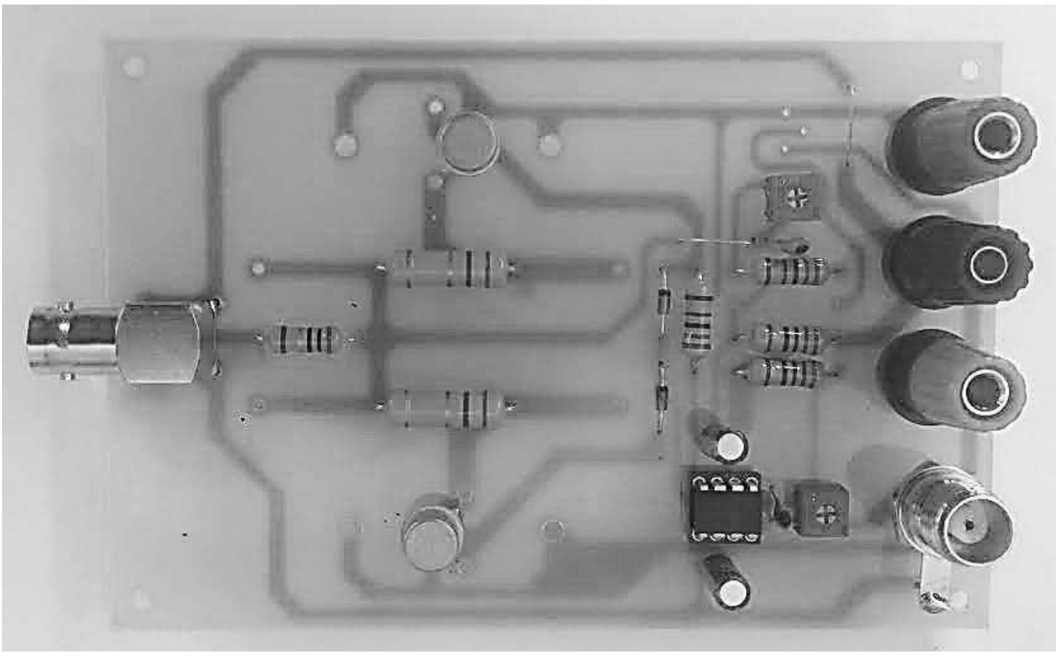


Figure 4.3 The circuit implementation on board of the preamplifier circuit.

Table 4.1 Summary of preamplifier circuit parameters and part numbers.

Preamplifier Circuit Parameters					
Resistors		**Capacitors**		**Integrated Circuits**	
Parts	**Values**	**Parts**	**Values**	**Parts**	**Numbers**
R_1	2kΩ	C_1	2pF	Op-Amp	CA3140
R_2	200Ω	C_2	50µF	Diode D_1	1N914
R_3	2.2kΩ	C_3	50µF	Diode D_2	1N914
R_4	2.2kΩ	-	-	Transistor Q_1	2N3053
R_5	10kΩ	-	-	Transistor Q_2	2N4037
R_6	3kΩ	-	-	-	-
R_7	200Ω	-	-	-	-
R_8	2.2kΩ	-	-	-	-
R_9	50kΩ	-	-	-	-
R_{10}	2.7Ω	-	-	-	-
R_{11}	2.7Ω	-	-	-	-
R_{12}	51Ω	-	-	-	-

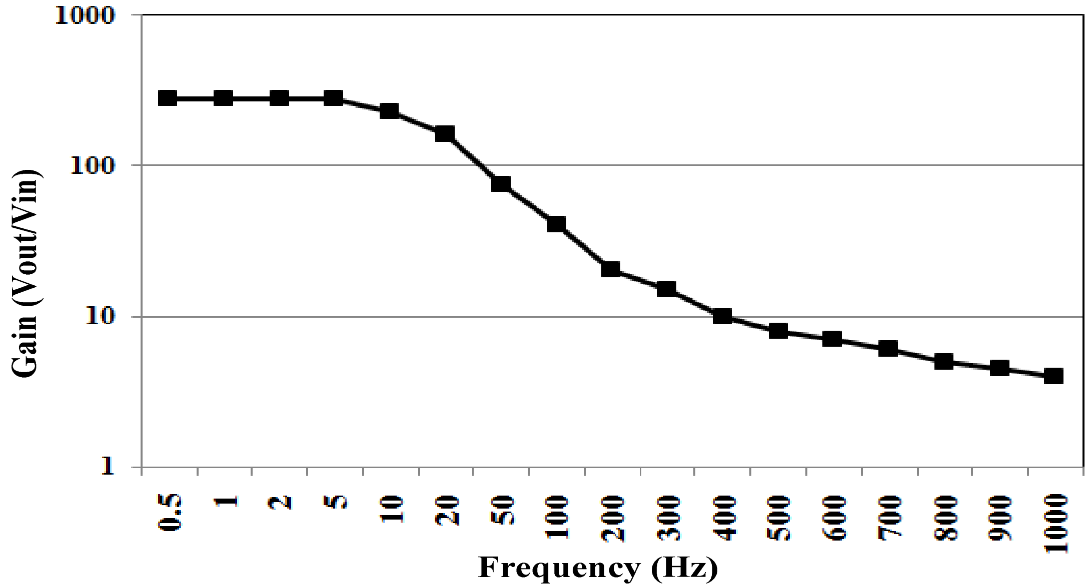


Figure 4.4 Frequency response characteristics of the proposed preamplifier.

Values of electrical components are listed as Table 4.1. The experimental results have been performed through the protoboard corresponding to the block diagram in Figure 4.2. The demonstration condition is a sinusoidal signal input of 20mV at 100 kHz, corresponding to the low-frequency signals. Figure 4.4 shows frequency characteristics of the proposed preamplifier. It can be seen from the graph that the preamplifier establish the low pass filter characteristic where the low frequency has a very high gain of above 100 but at the high frequency, for example 1 MHz, the gain is still greater than 0. For the frequency of interest at 100 kHz the gain is approximately at 20dB. Figure 4.5 illustrates the sinusoidal signal in time domain. It can be seen that the non-inverting amplifier can amplify the input signal to the larger signal without phase shift.

Figure 4.6 illustrates of the Fast Fourier Transform (FFT) for Total Harmonic Distortion (THD) calculations. Using the rectangular window transformation, the peak signal at 100 kHz has the highest peak where the noise floor is about 200 kHz far from the center frequency. Since FFT transformation was used for the THD calculation, Table 4.2 summarized the THD in percentage at difference value of particular gain under region of interest between 5-20. It is seen that the THD is in the range of 3-5%.

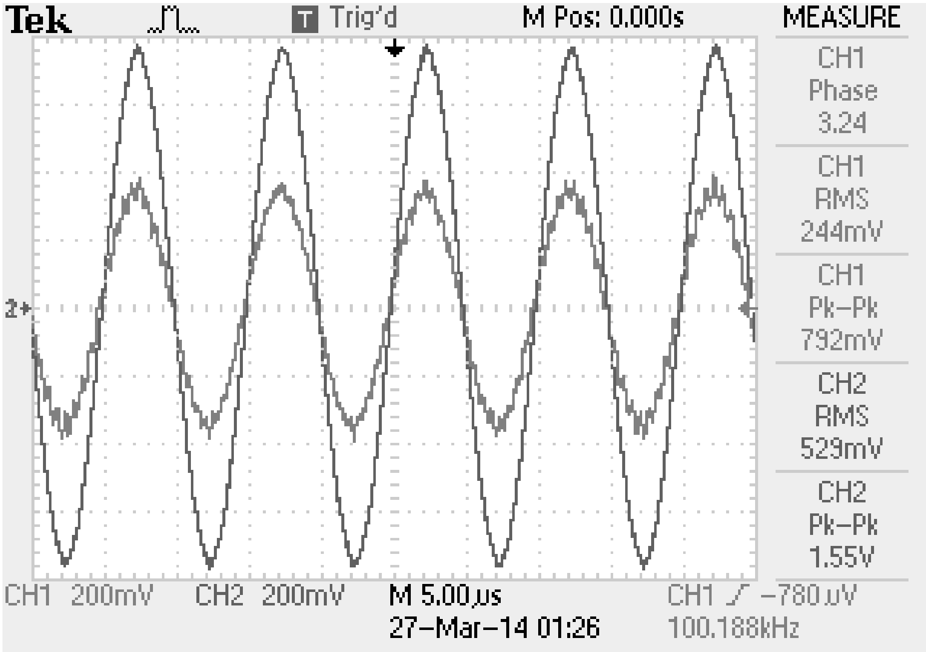


Figure 4.5 Illustration of sinusoidal signal in time domain.

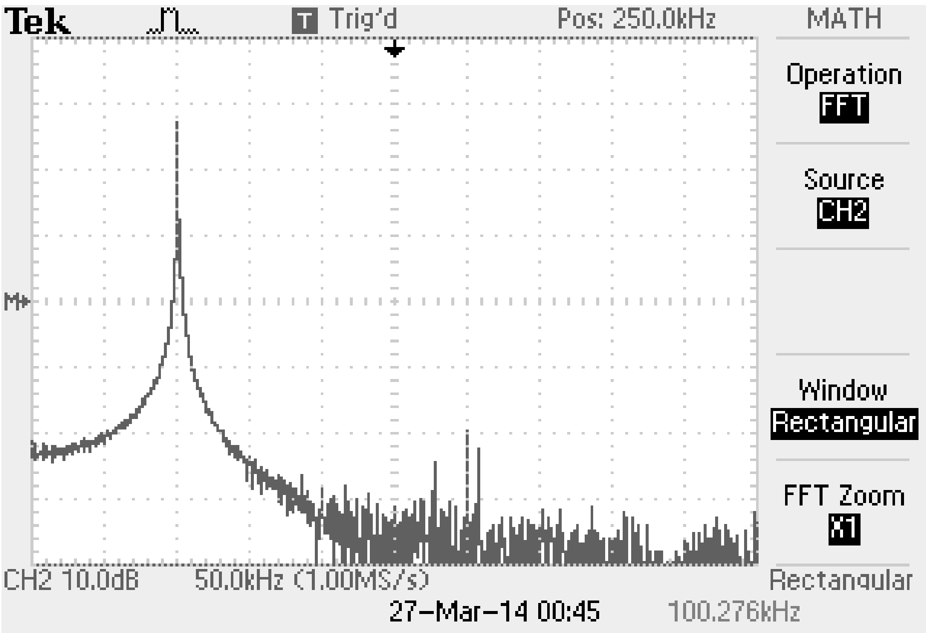


Figure 4.6 Illustration of the Fast Fourier Transform for THD calculations.

Table 4.2 Summary of particular gains (V_{out}/V_{in}) and showing the corresponding THD in percentages

Particular Gains	THD (%)
5	3.2362
10	3.7334
15	5.4146
20	5.6488

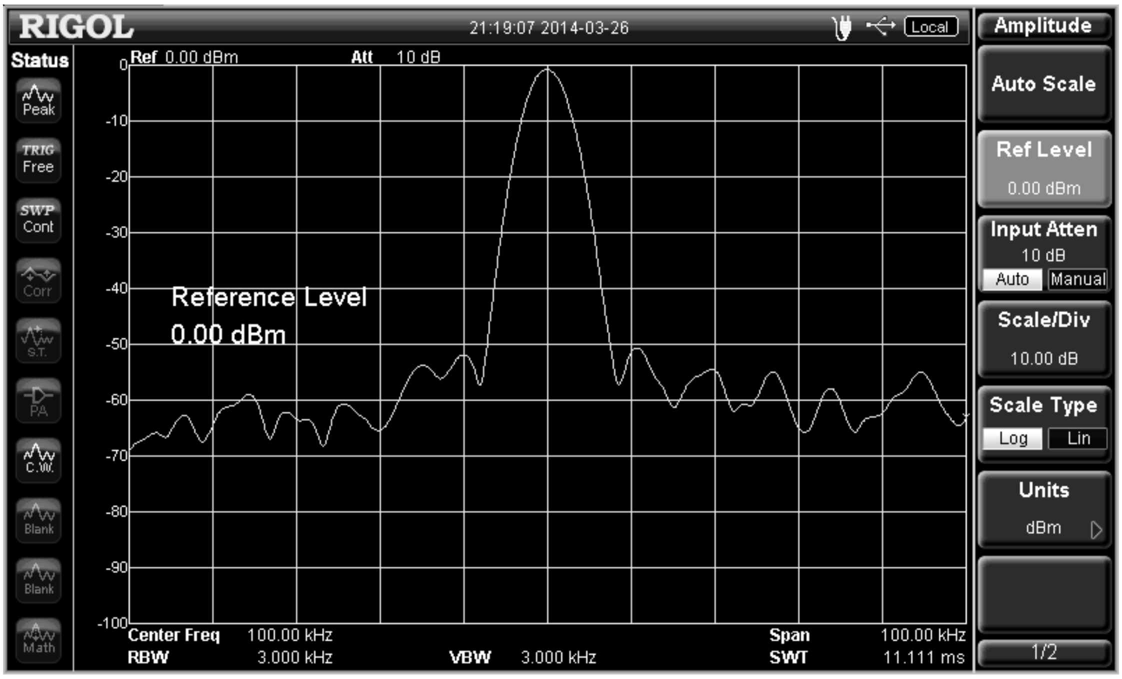


Figure 4.7 Illustration of the power spectral density showing the bandwidth of the signal at 0dBm level.

This is acceptable for the preamplifier that amplifies the AE signal in real application. Figure 4.7 illustrates the power spectral density showing the bandwidth of the signal at 0dBm level. It is seen that the noise floor is approximately -60dB and the bandwidth is about 4 kHz.

4.3 The Design of STM32F4 Microcontroller for A/D Converter Operations and MATLAB for GUI

As mentioned earlier, this research utilizes the STM32F4 controller, which is based on the high-performance ARM®Cortex™-M4 32-bit RISC core operating at a frequency of up to 168 MHz, for analog-to-digital conversion. This device offers three 12-bit ADCs, two DACs, a low-power RTC, twelve general-purpose 16-bit timers including two PWM timers for motor control, two general-purpose 32-bit timers. Therefore, the input analog signal is digitized with the performance of 3×12-BIT AMD 2.4 MSPS. Figures 4.8 and 4.9 demonstrate experimental setup of the STM32F4 and the GUI that exploits MATLAB program to display real-time AE signals and various signal characteristics analysis.

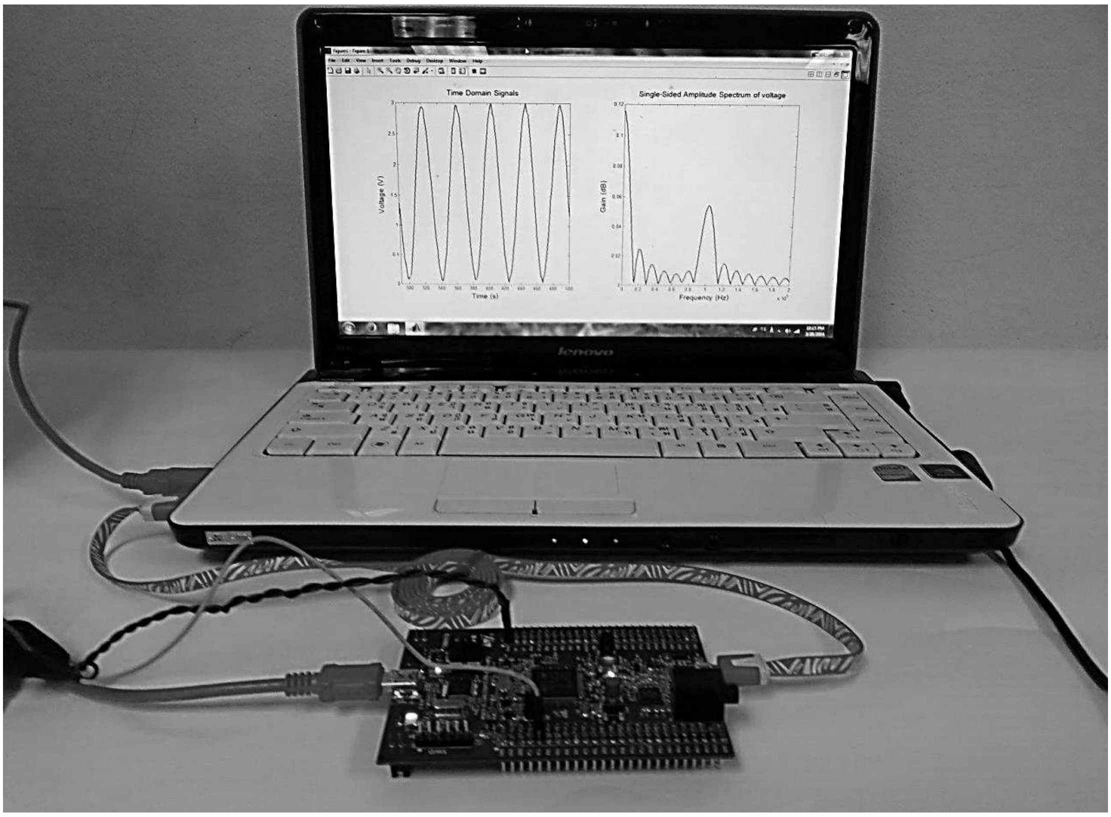


Figure 4.8 Demonstration of the experimental setup of STM32F4 and the GUI.

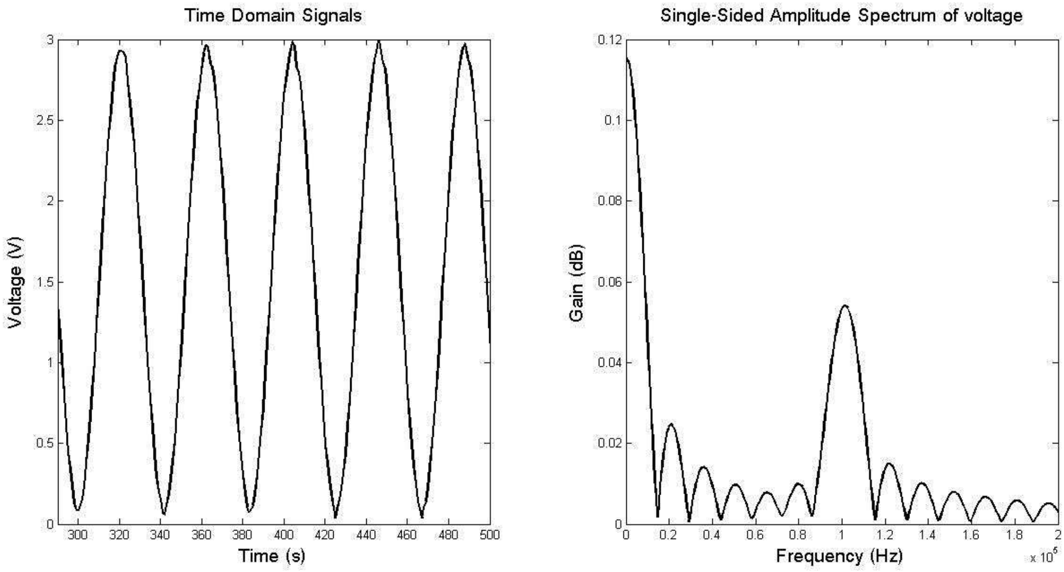


Figure 4.9 Demonstration of the display of signal in time and frequency domains.

4.4 The Design of Xbee for Wireless Communications

In order to transmit AE signal wirelessly, XBee wireless module is programmed. The first module connects with STM32F4 microcontroller, which is so called transceiver (Tx) and the another module connects with personal computer via USB port, which is so called receiver (Rx). Figures 4.10 and 4.11 illustrate the experimental setup of wireless transceiver and Figure 4.12 and 4.13 also illustrate the experimental setup of wireless receiver.

XBee wireless module is performed base on IEEE 802.15.4, which is so called ZigBee technology. This technology is a low data rate, low power consumption and low cost wireless networking protocol. Refer to the specification, XBee wireless module can transmit data in the range of 120 m (line-of-sight) with 2mW output at data rate 250kbps. Figure 4.14 demonstrates the experimental result when transmit data 00 and FF from STM32F4 microcontroller to a personal computer via XBee wireless modules in the range of 20 m. The results show that data transmission via XBee wireless modules is acceptable for the proposed system.

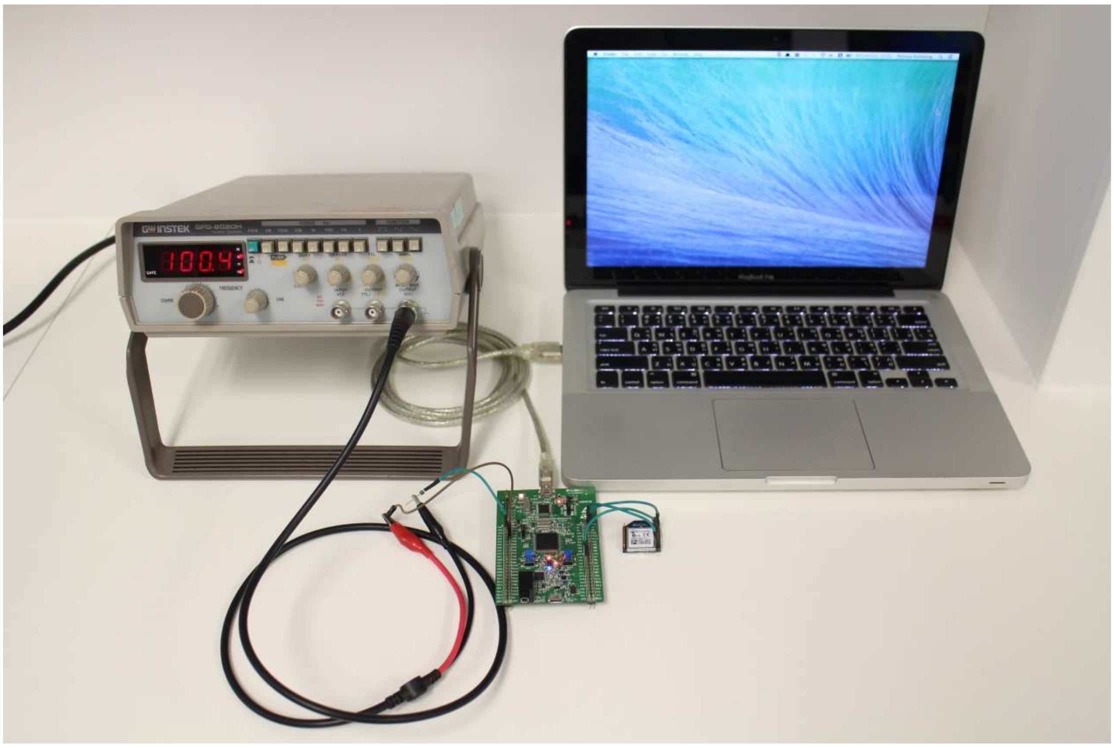


Figure 4.10 The experimental setup of XBee Transceiver.

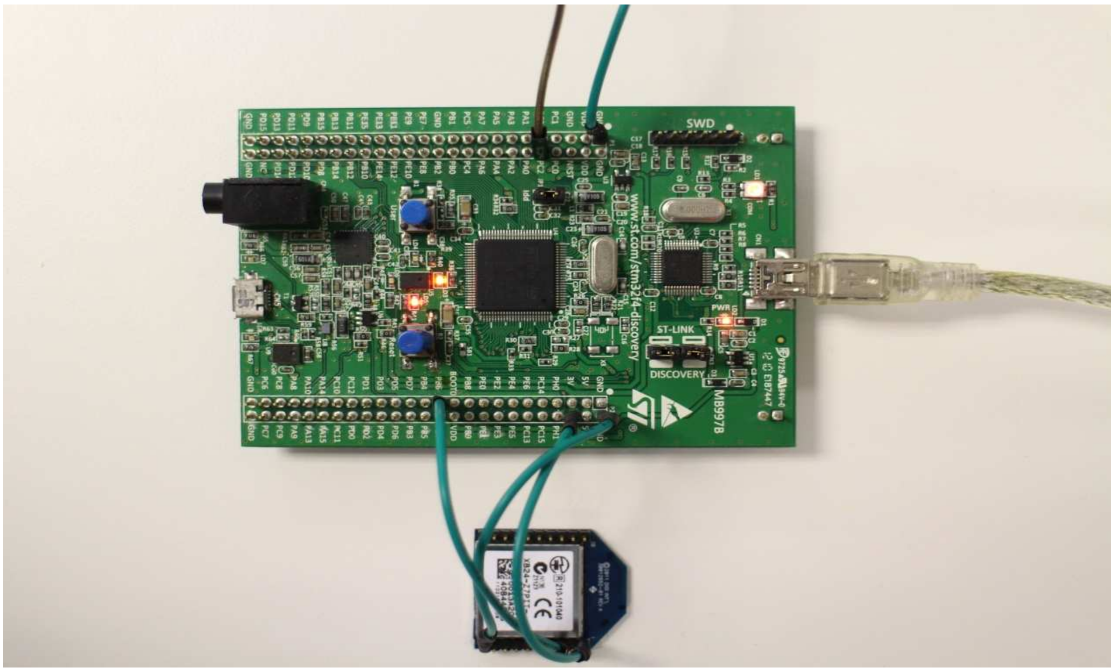


Figure 4.11 The connection of STM32F4 and XBee for Transceiver.

Figure 4.12 The experimental setup of XBee Receiver.

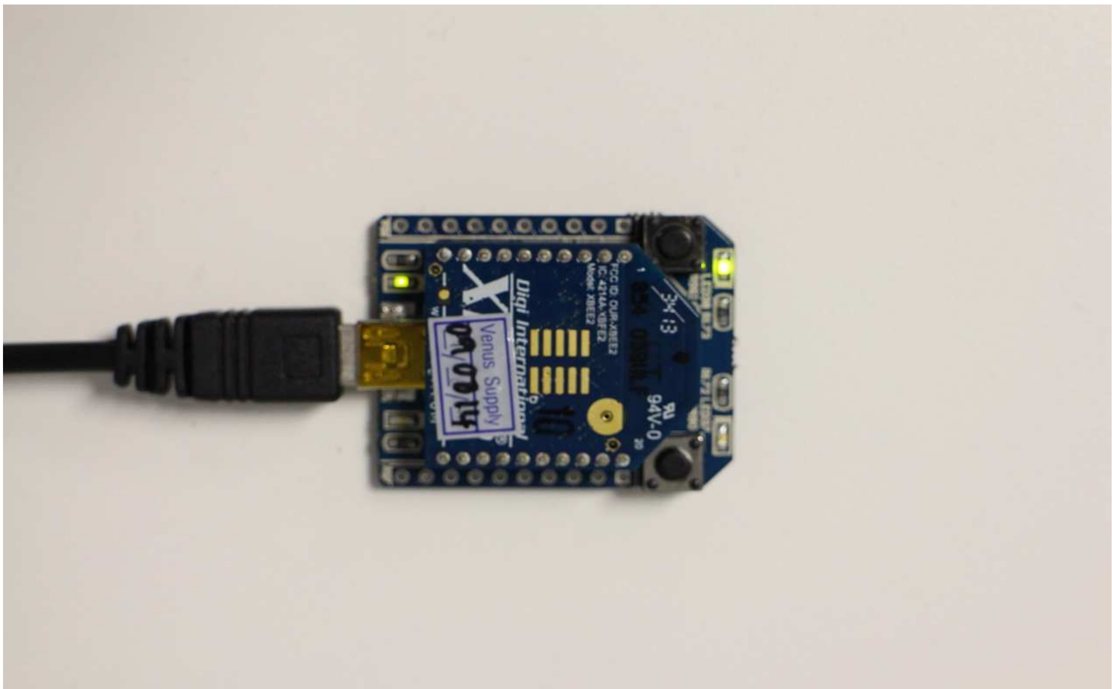


Figure 4.13 Illustration of XBee Receiver.

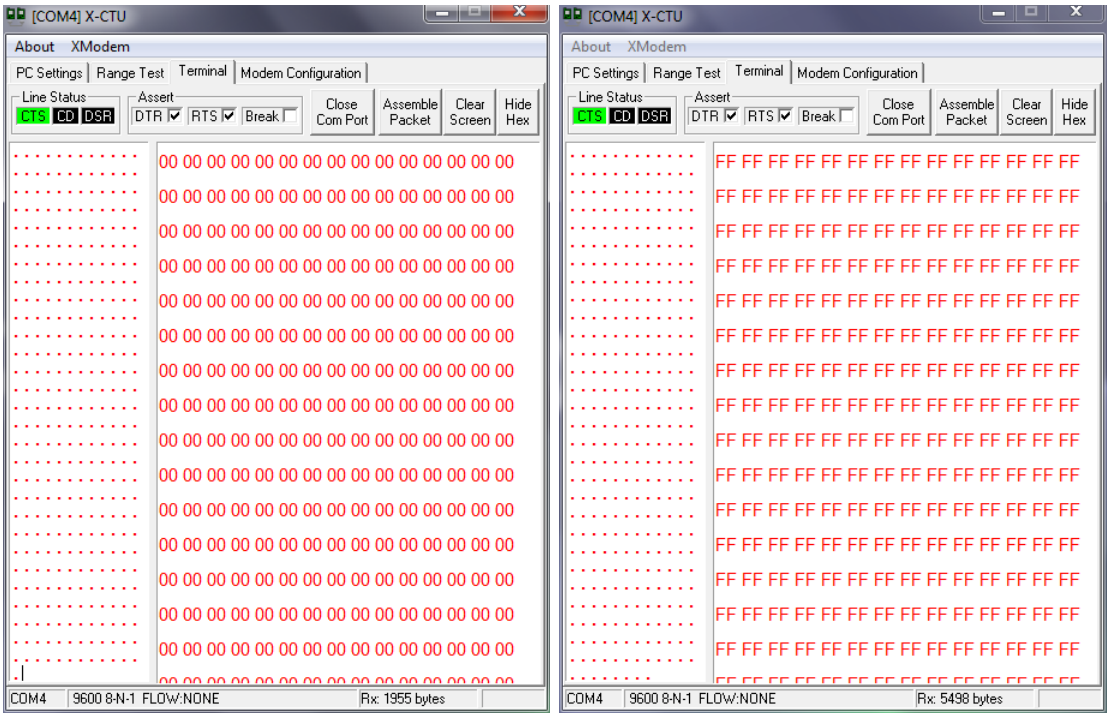


Figure 4.14 The experimental result when transmit data 00 and FF from STM32F4 microcontroller to a personal computer via XBee.

4.5 Artificial Sinusoidal Signal Tests using LabVIEW

In order to verify signal analysis of LabVIEW particularly in the Fast Fourier Transform (FFT) function, the artificial sinusoidal signal tests are performed. Figures 4.15 to 4.28 show the waveforms of the artificial sinusoidal signal in various frequency in time domain and the corresponding power spectral density in dB, respectively. The experimental results show that LabVIEW program can be utilized as a verified tool for the proposed system.

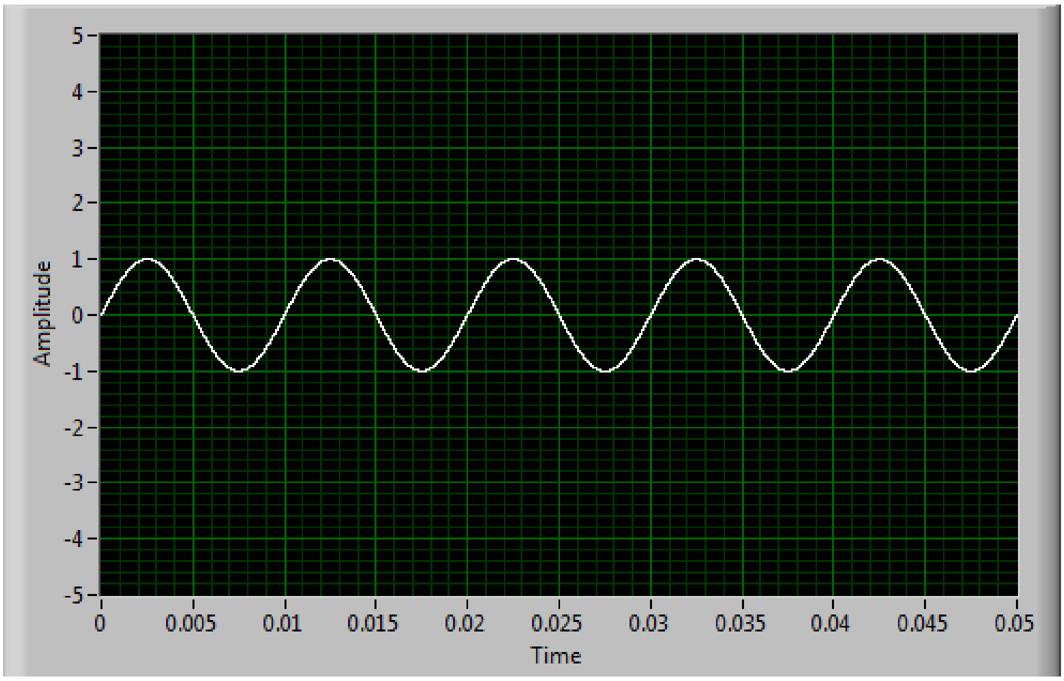


Figure 4.15 The waveform of an artificial sinusoidal signal in LabVIEW at 100 Hz.

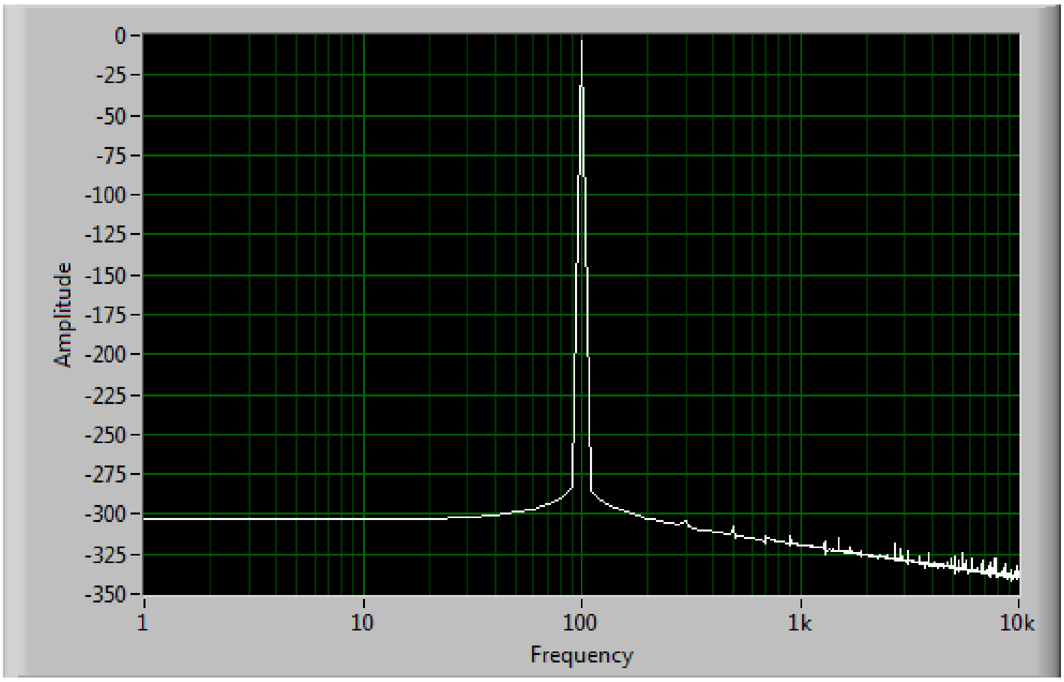


Figure 4.16 The corresponding power spectral density in dB of the waveform in Figure 4.15 using LabVIEW.

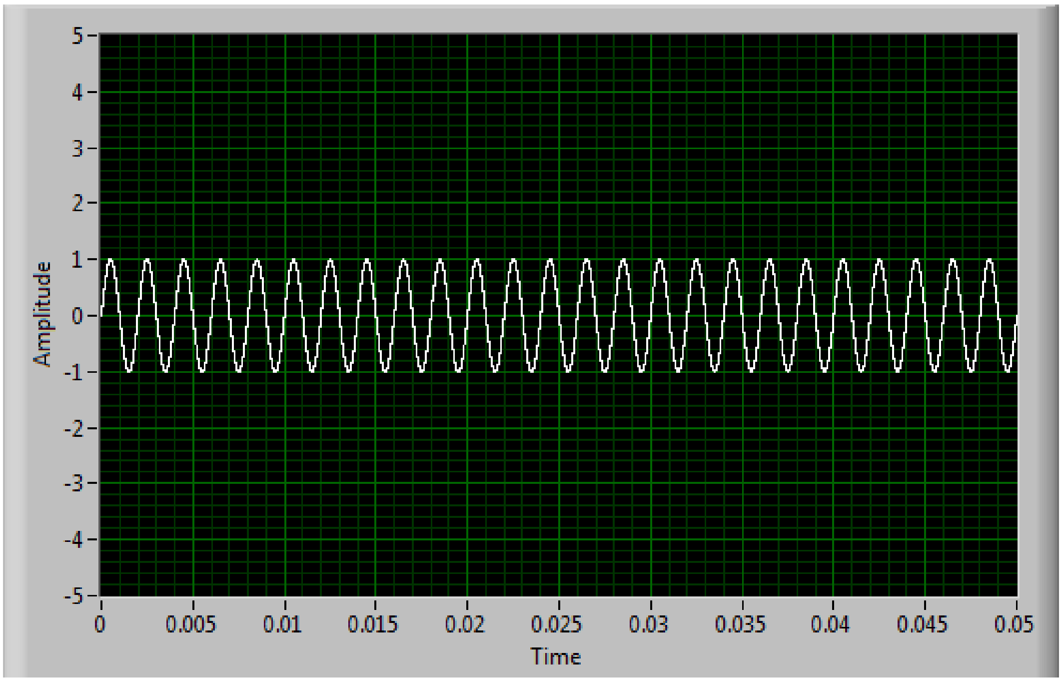


Figure 4.17 The waveform of an artificial sinusoidal signal in LabVIEW at 500 Hz.

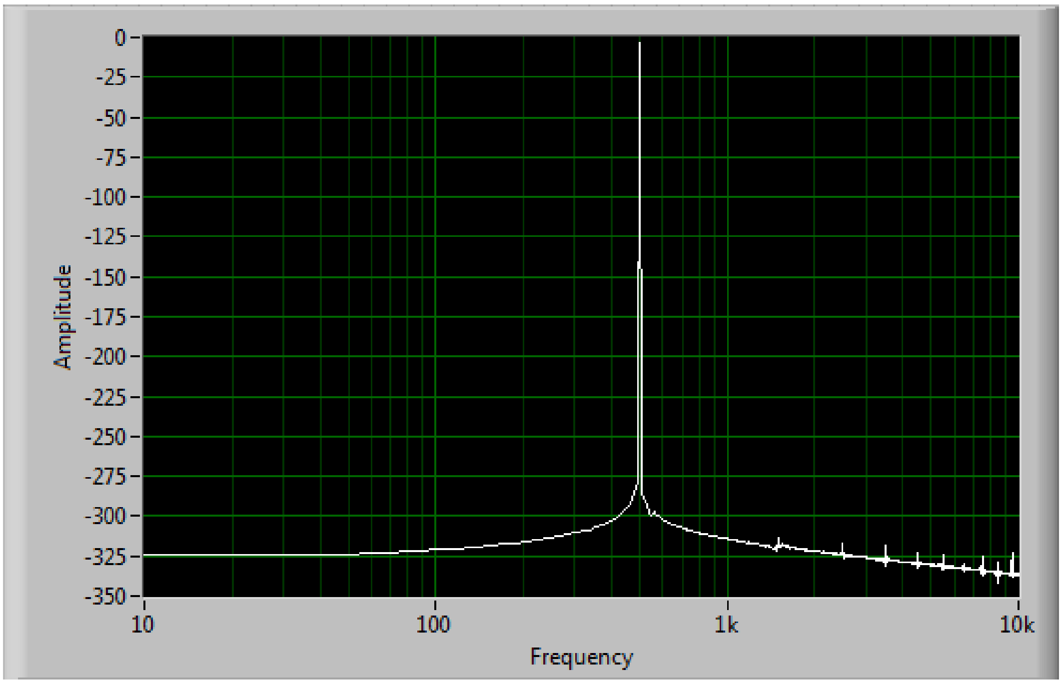


Figure 4.18 The corresponding power spectral density in dB of the waveform in Figure 4.17 using LabVIEW.

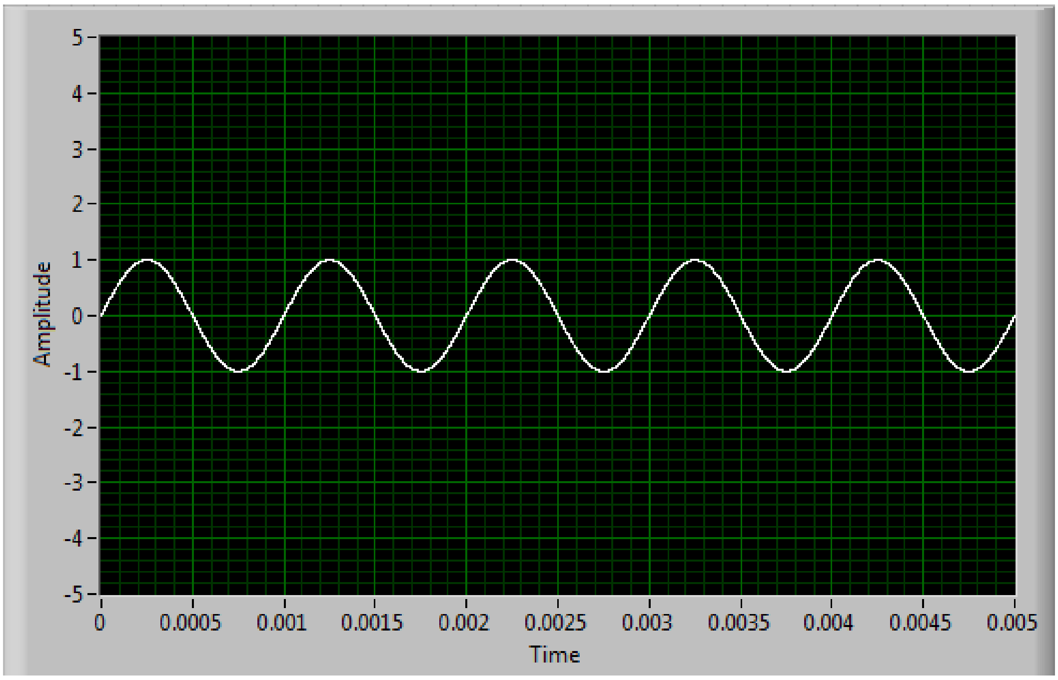


Figure 4.19 The waveform of an artificial sinusoidal signal in LabVIEW at 1 kHz.

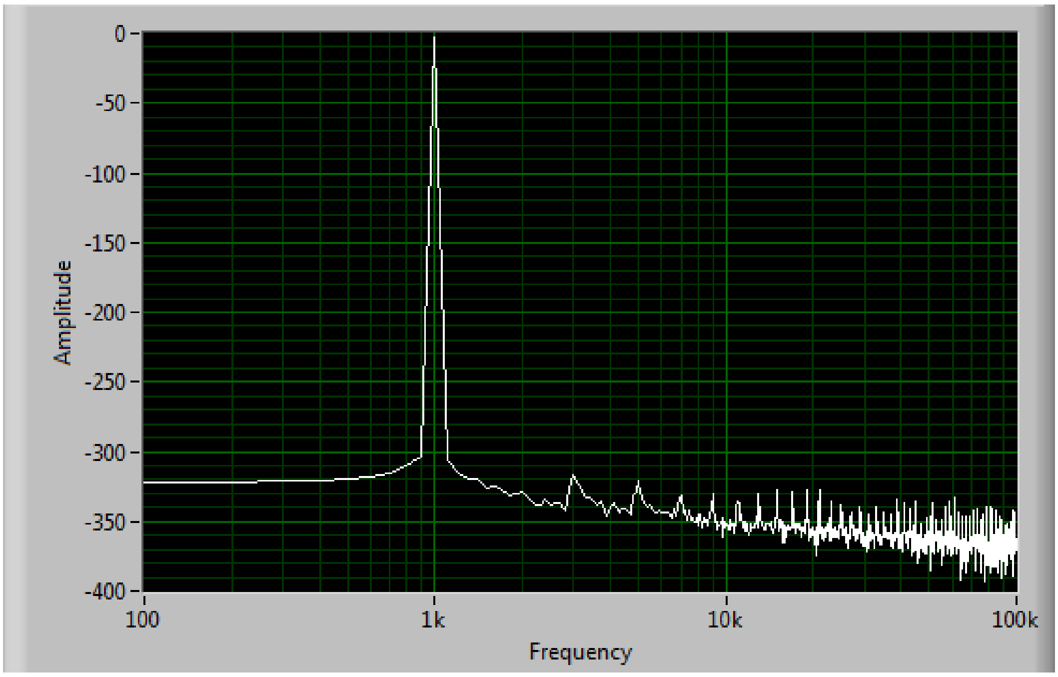


Figure 4.20 The corresponding power spectral density in dB of the waveform in Figure 4.19 using LabVIEW.

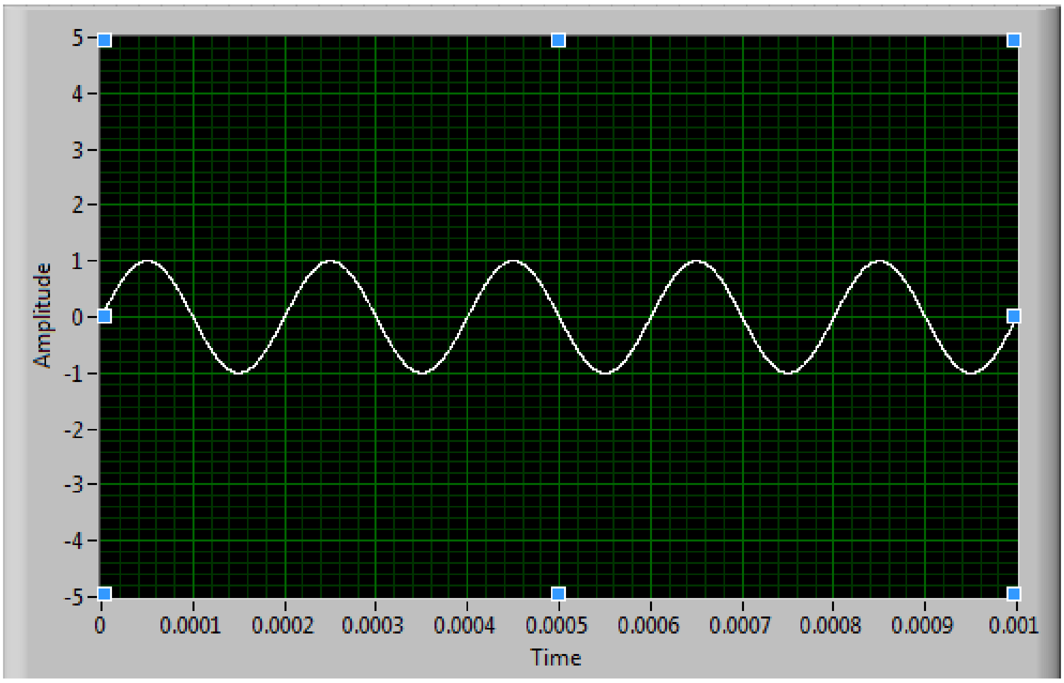


Figure 4.21 The waveform of an artificial sinusoidal signal in LabVIEW at 5 kHz.

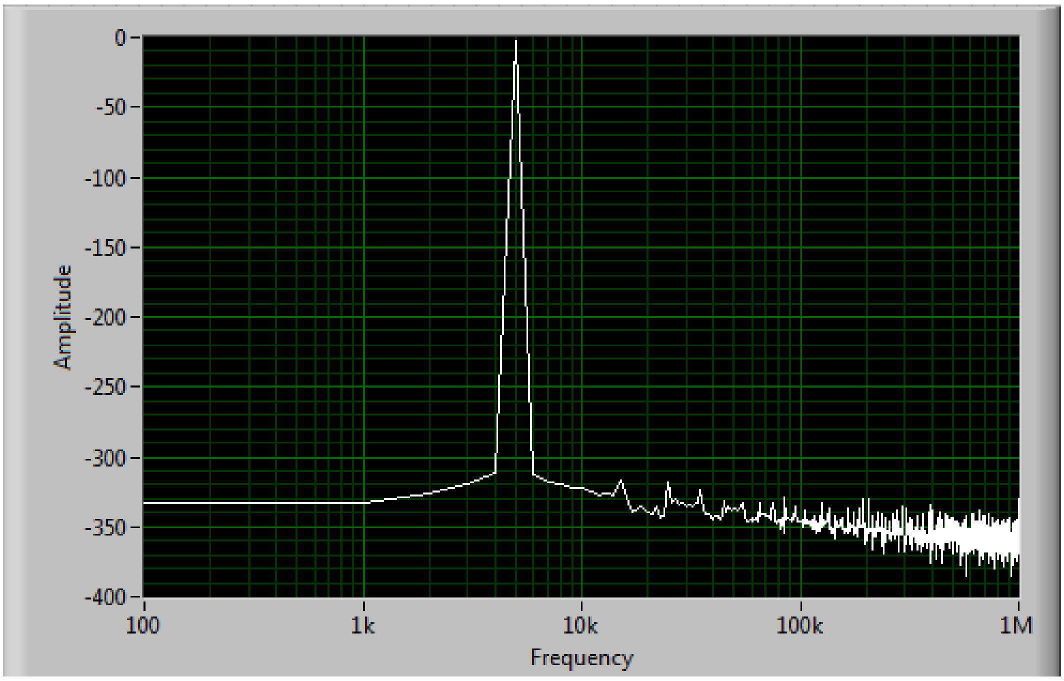


Figure 4.22 The corresponding power spectral density in dB of the waveform in Figure 4.21 using LabVIEW.

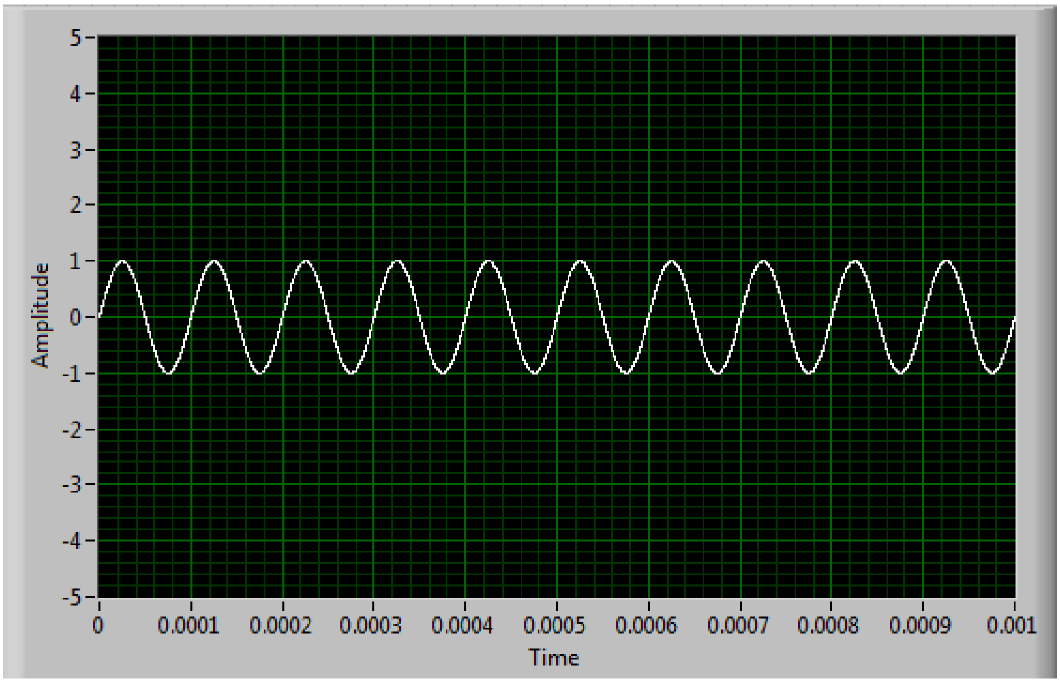


Figure 4.23 The waveform of an artificial sinusoidal signal in LabVIEW at 10 kHz.

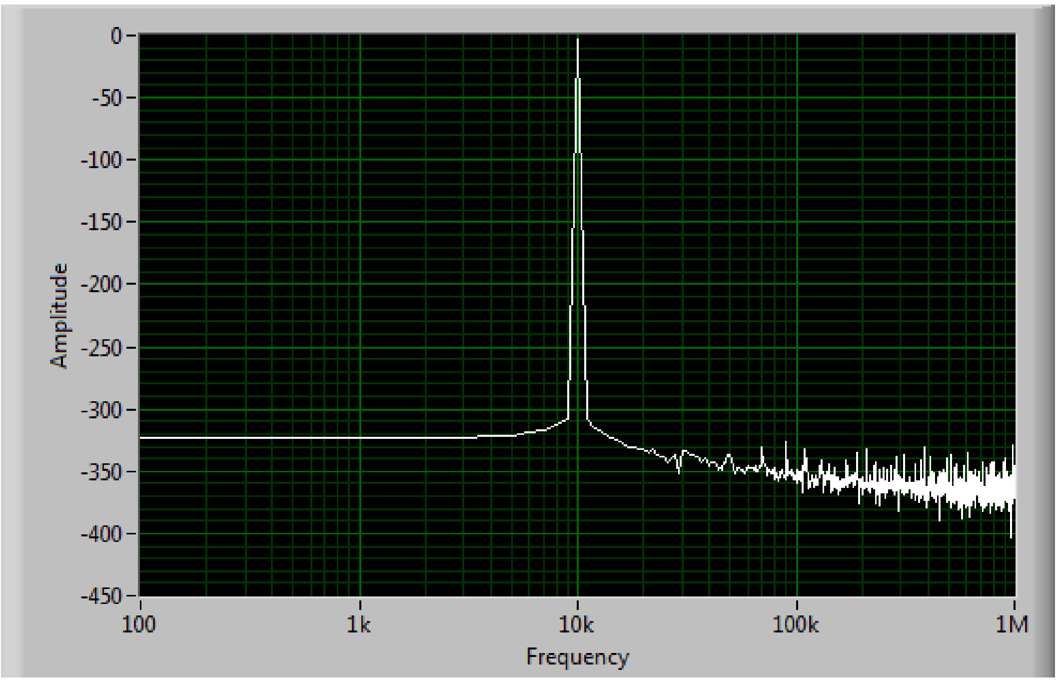


Figure 4.24 The corresponding power spectral density in dB of the waveform in Figure 4.23 using LabVIEW.

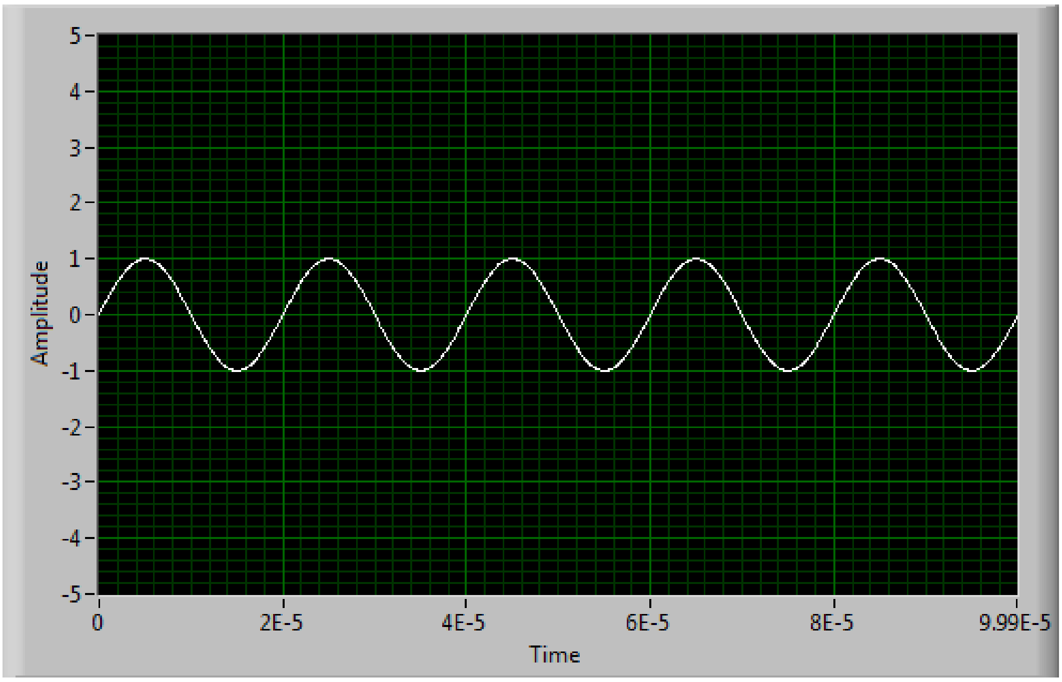


Figure 4.25 The waveform of an artificial sinusoidal signal in LabVIEW at 50 kHz.

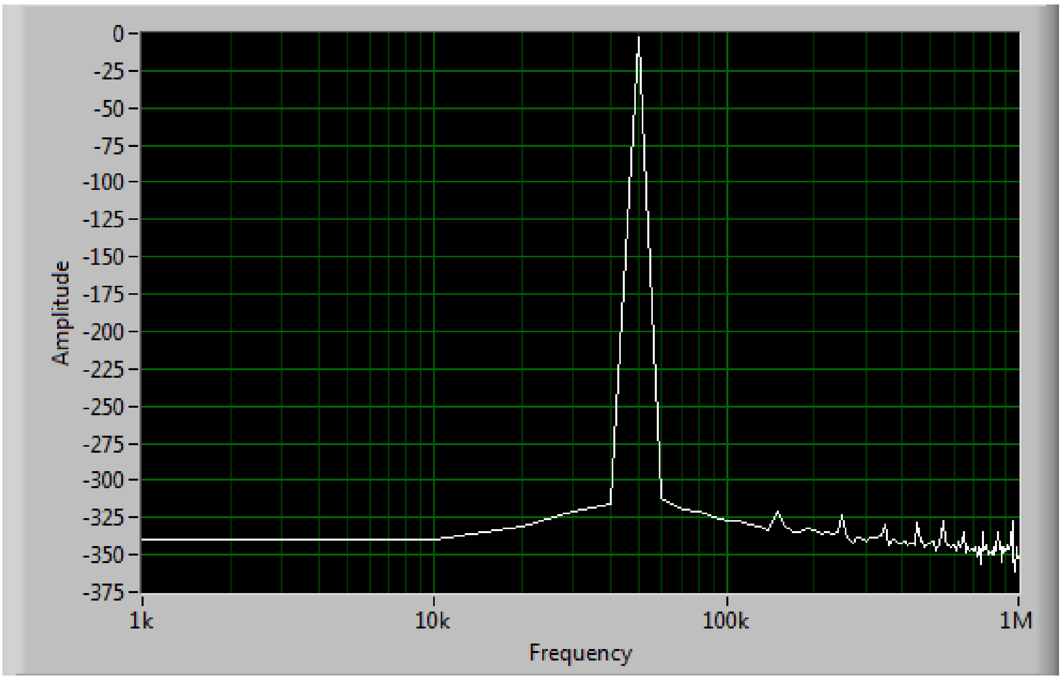


Figure 4.26 The corresponding power spectral density in dB of the waveform in Figure 4.25 using LabVIEW.

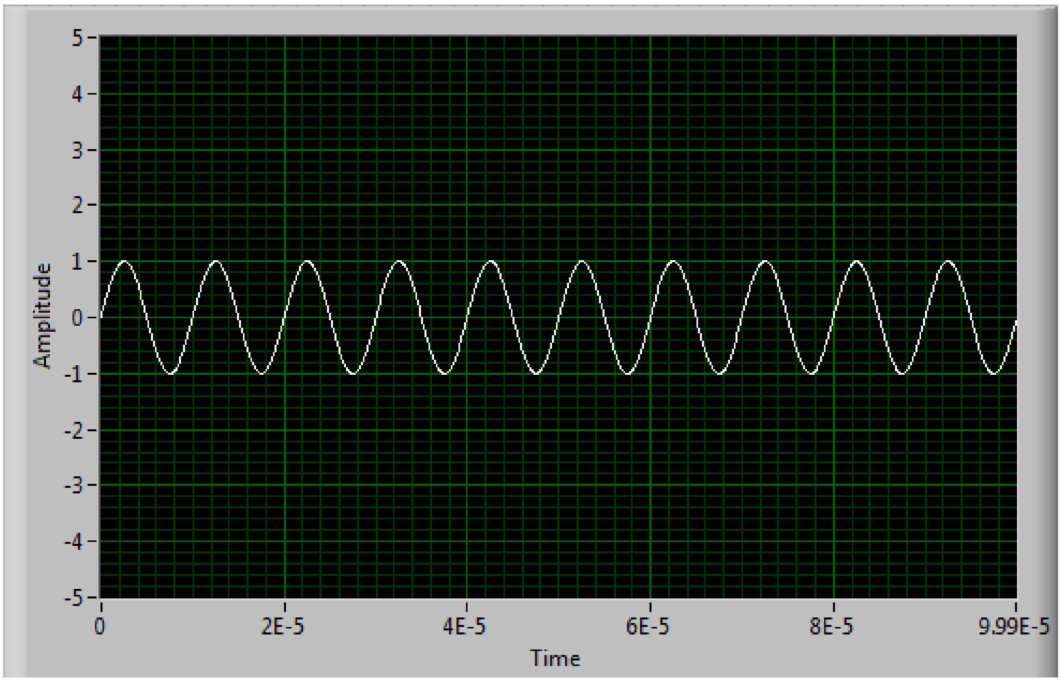


Figure 4.27 The waveform of an artificial sinusoidal signal in LabVIEW at 100 kHz.

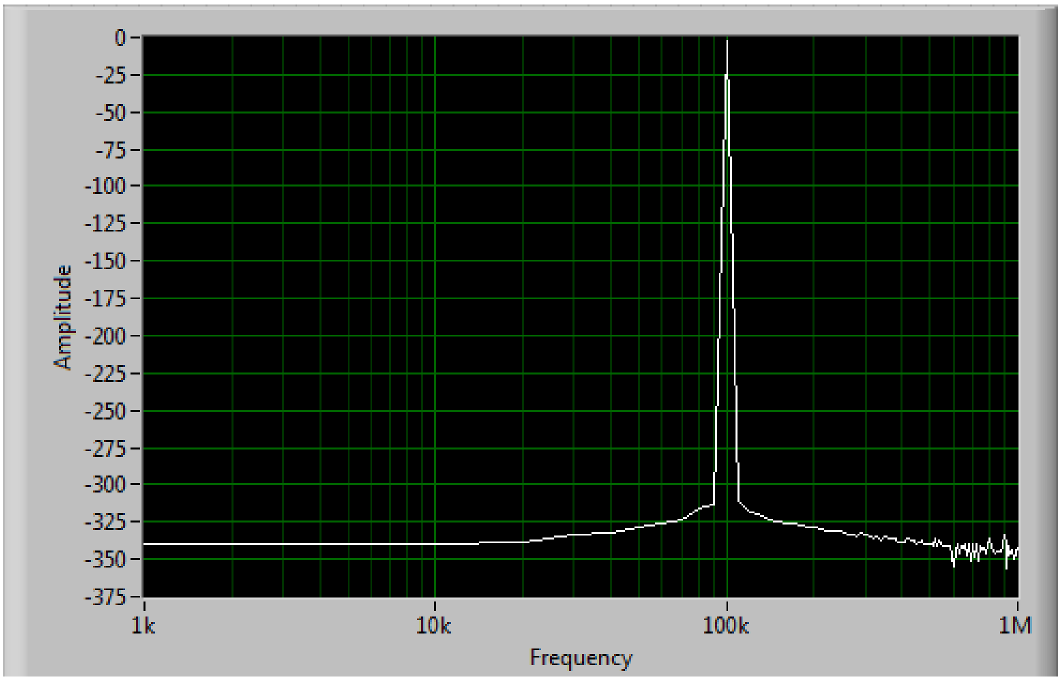


Figure 4.28 The corresponding power spectral density in dB of the waveform in Figure 4.27 using LabVIEW.

4.6 Real Sinusoidal Signal Tests using LabVIEW and MATLAB

In this test, the comparison between the commercially DAQ board using the GUI in LabVIEW and the proposed DAQ system by STM32F4 using the GUI in MATLAB as shown in Figure 4.29 have been performed. The real sinusoidal signal in the frequency range of 100 Hz – 100 kHz adjusted by a function generator is applied to each system. The waveforms and corresponding power spectral density in dB from the commercially system are shown as Figures 4.30 to 4.39 and from the proposed system are also shown as Figures 4.40 to 4.49. The experimental results show that the DAQ system by STM32F4 and the GUI in MATLAB is acceptable for the proposed system.

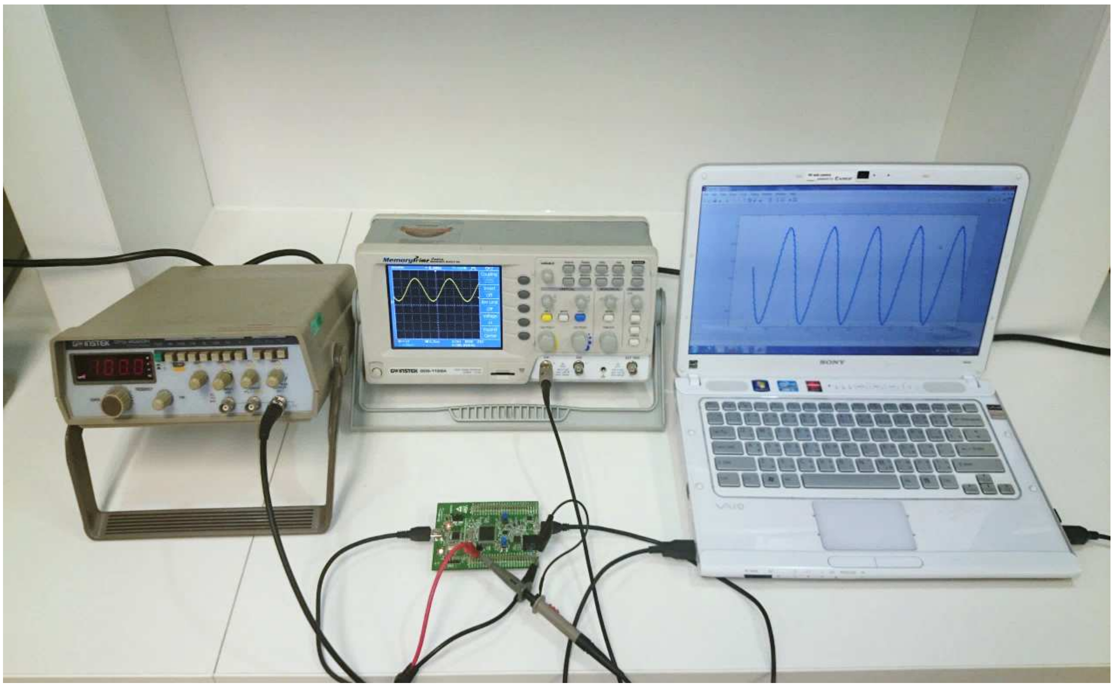


Figure 4.29 The experimental set up of real sinusoidal signal test using MATLAB.

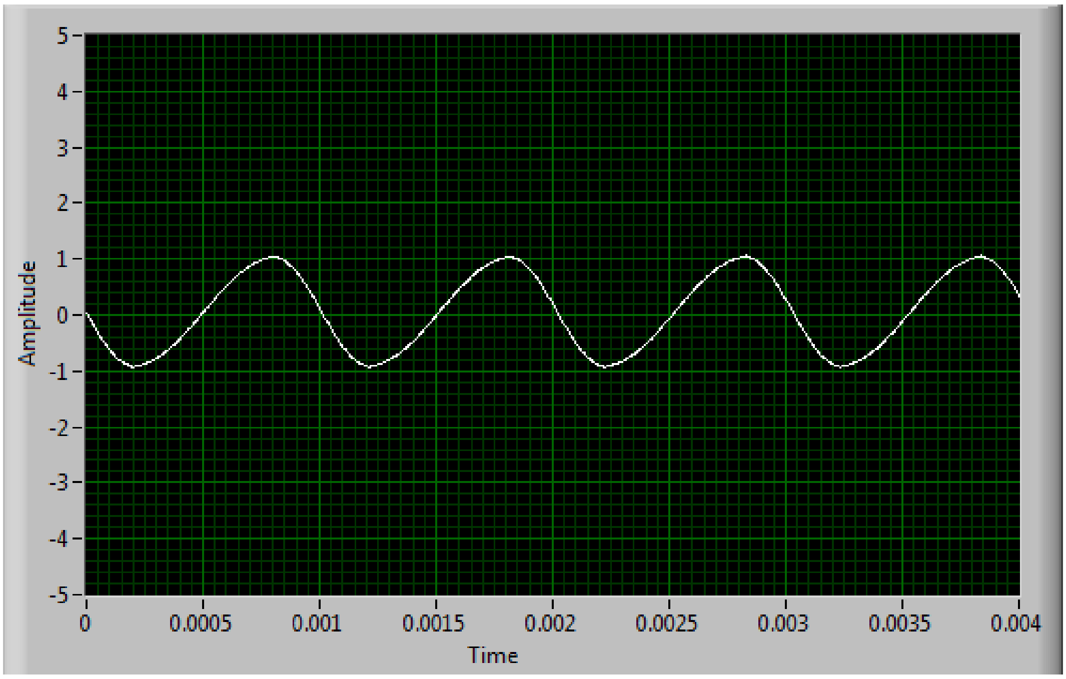


Figure 4.30 The waveform of real sinusoidal signal in LabVIEW at 1 kHz.

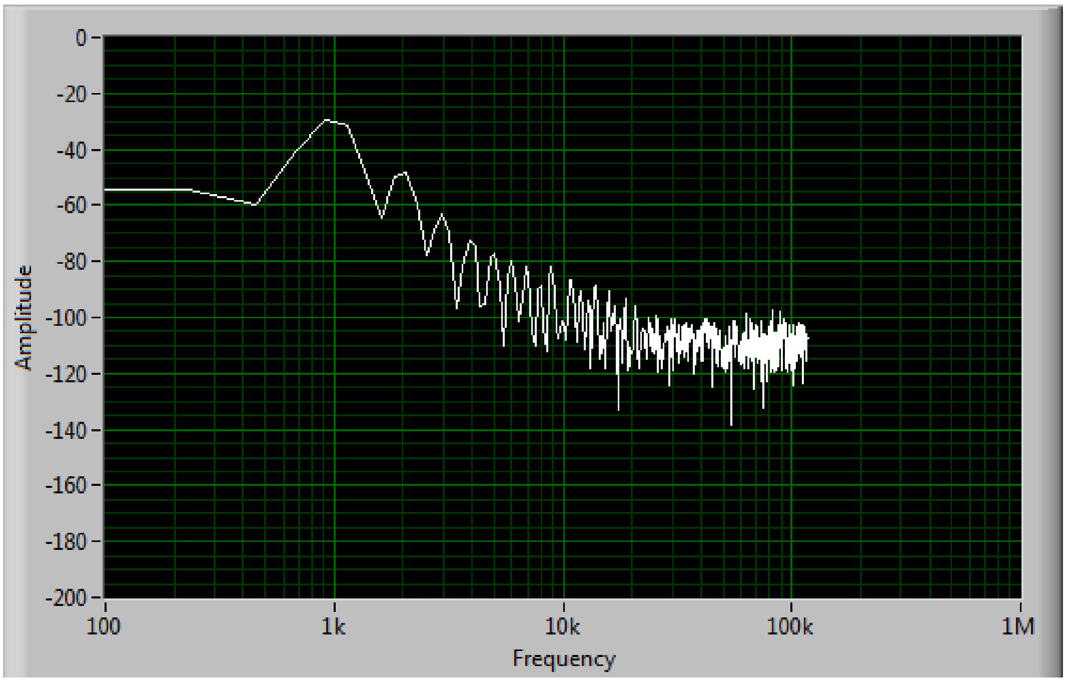


Figure 4.31 The corresponding power spectral density in dB of the waveform in Figure 4.30 using LabVIEW.

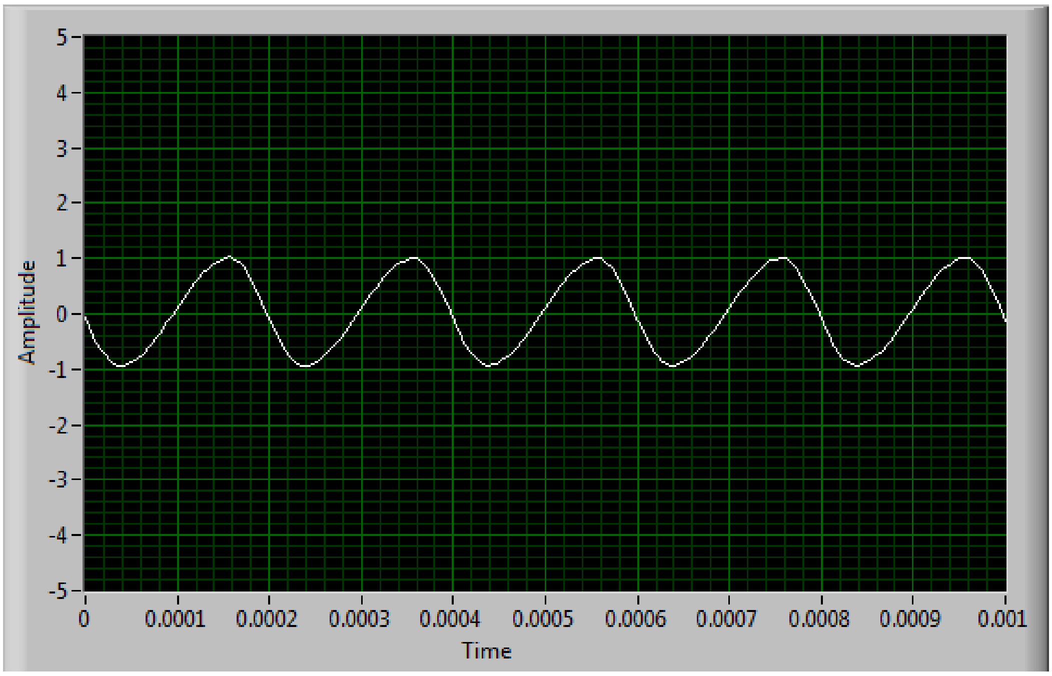


Figure 4.32 The waveform of real sinusoidal signal in LabVIEW at 5 kHz.

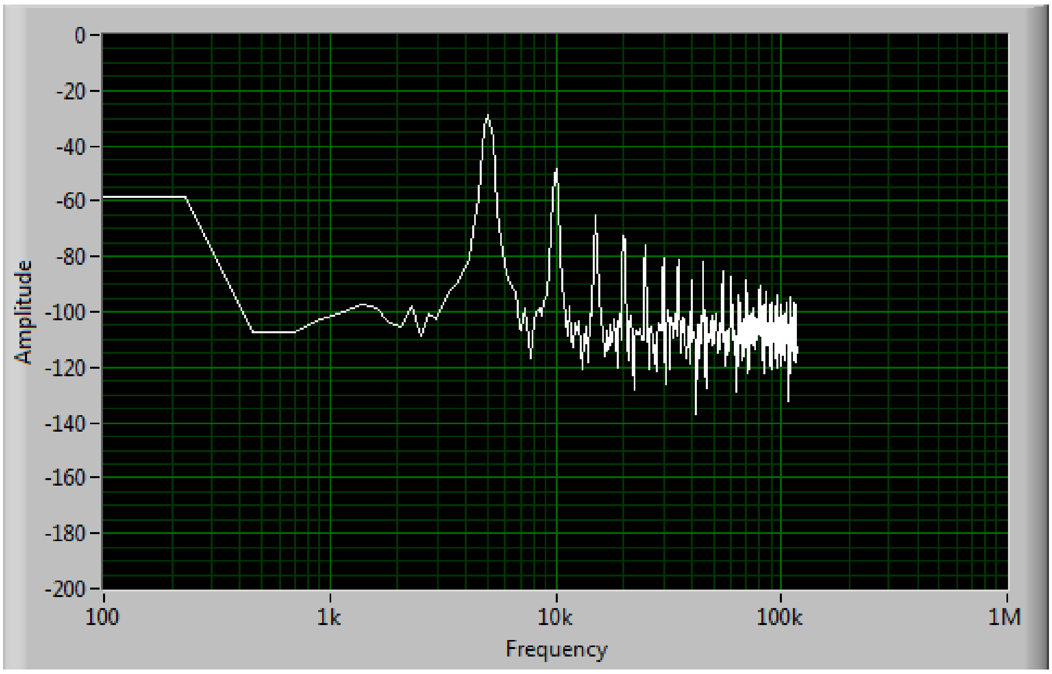


Figure 4.33 The corresponding power spectral density in dB of the waveform in Figure 4.32 using LabVIEW.

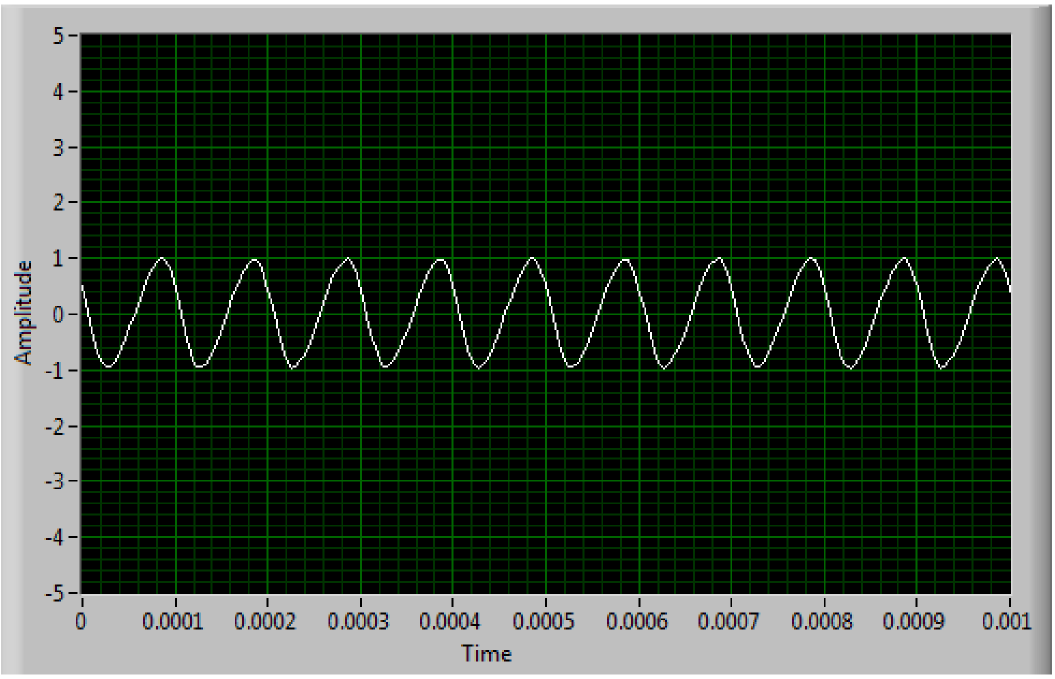


Figure 4.34 The waveform of real sinusoidal signal in LabVIEW at 10 kHz.

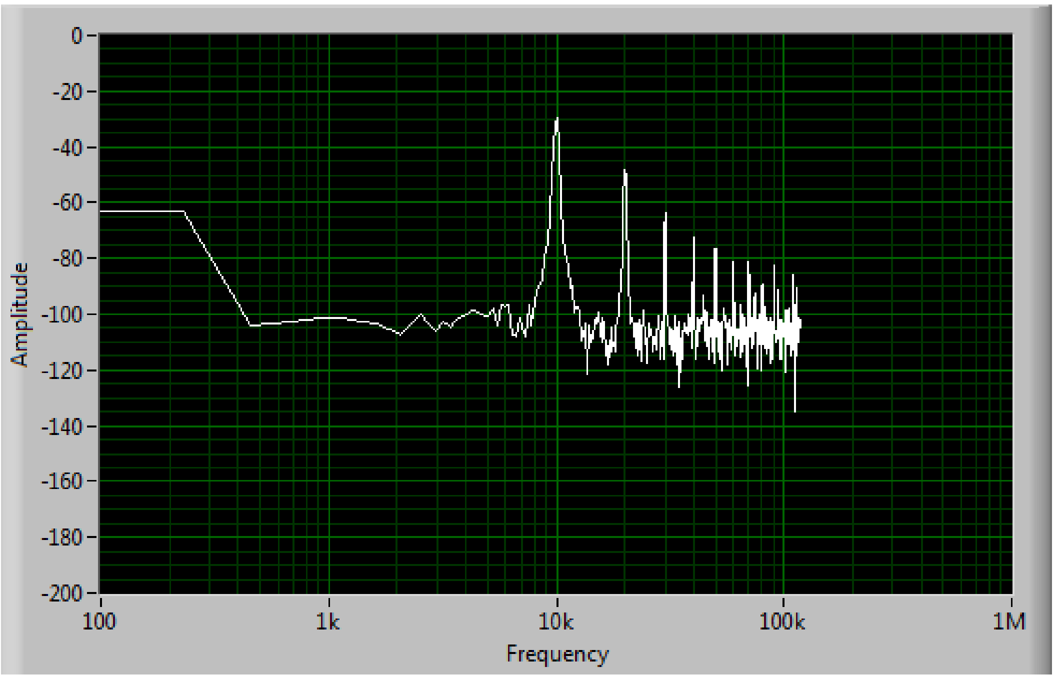


Figure 4.35 The corresponding power spectral density in dB of the waveform in Figure 4.34 using LabVIEW.

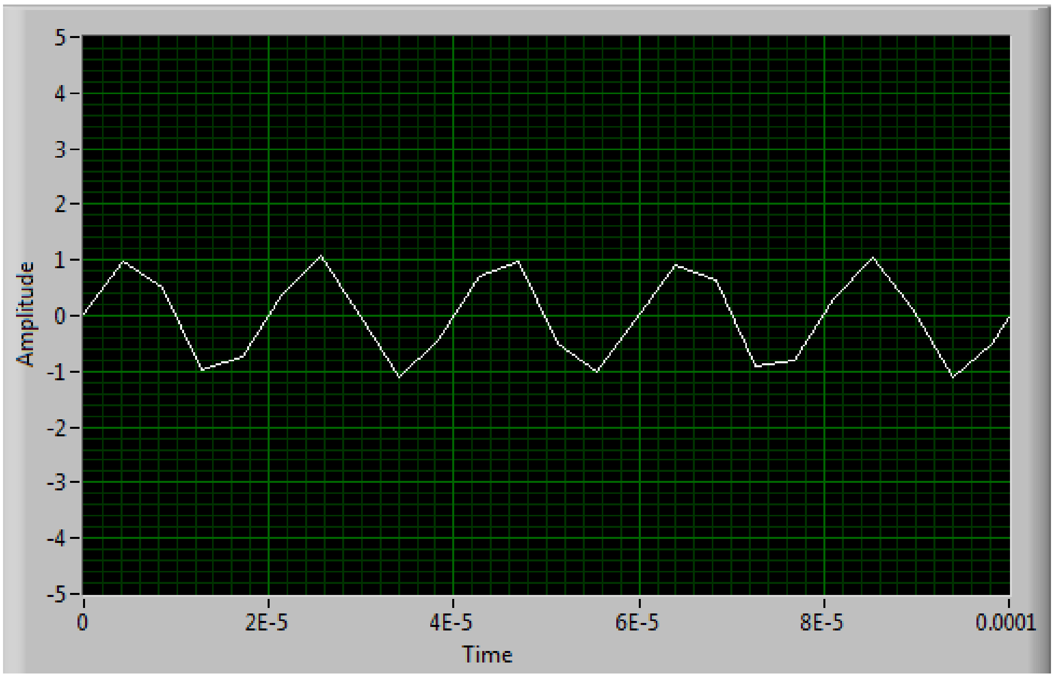


Figure 4.36 The waveform of real sinusoidal signal in LabVIEW at 50 kHz.

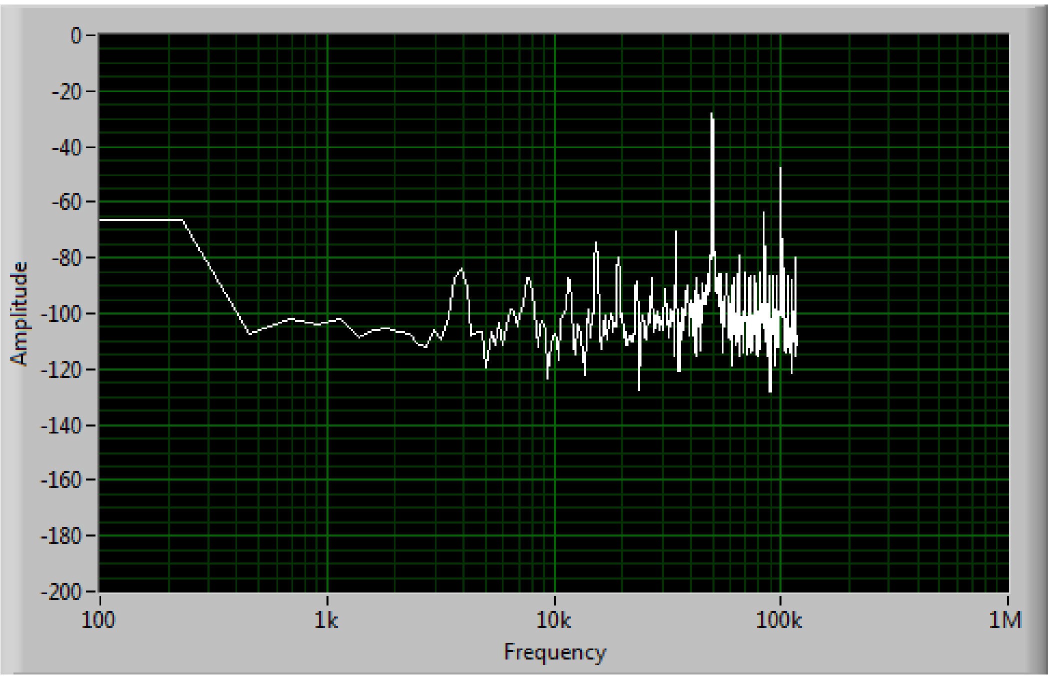


Figure 4.37 The corresponding power spectral density in dB of the waveform in Figure 4.36 using LabVIEW.

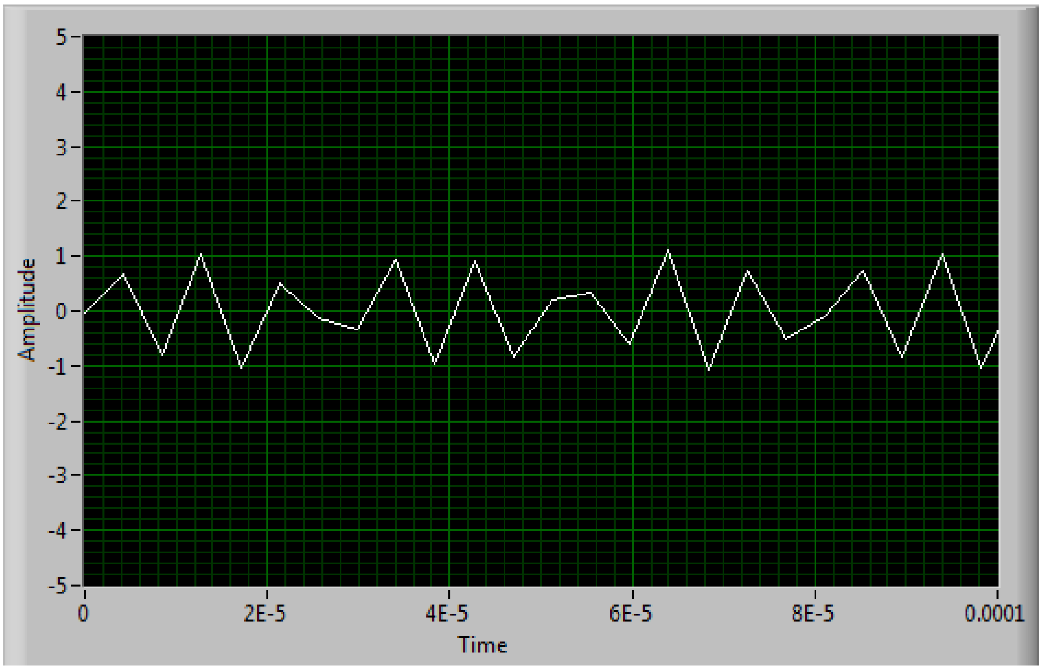


Figure 4.38 The waveform of real sinusoidal signal in LabVIEW at 100 kHz.

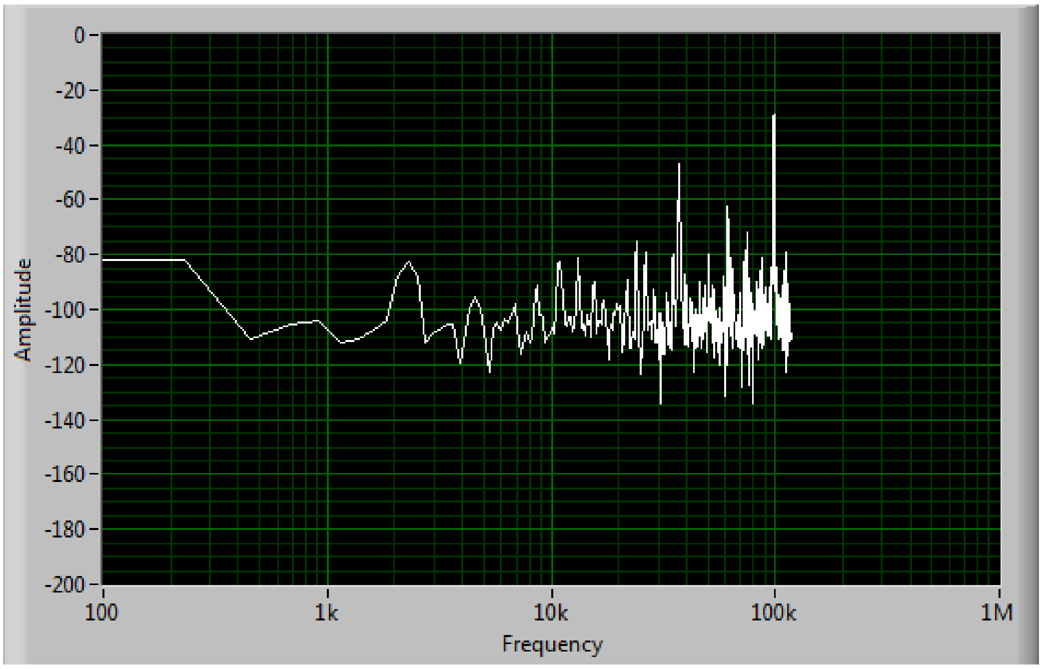


Figure 4.39 The corresponding power spectral density in dB of the waveform in Figure 4.38 using LabVIEW.

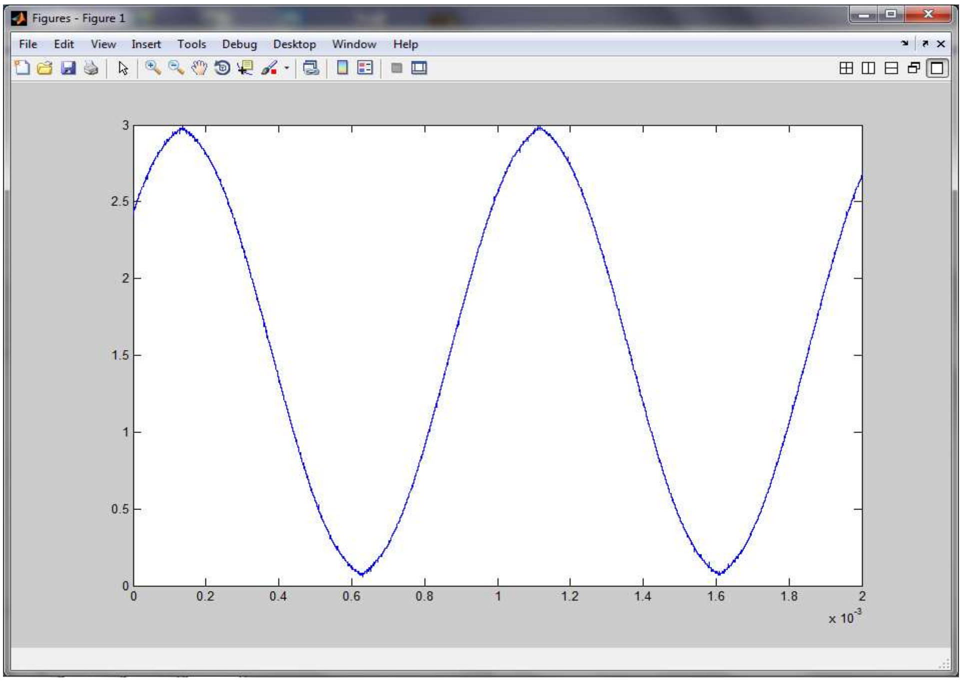


Figure 4.40 The waveform of sinusoidal signal at 1 kHz from a function generator in MATLAB.

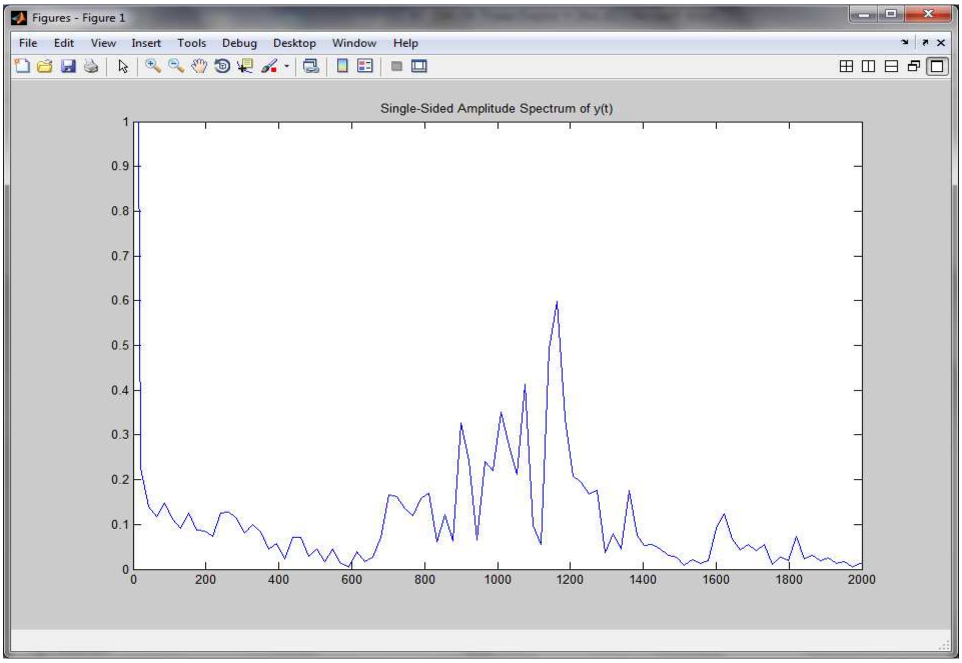


Figure 4.41 The corresponding power spectral density in dB of the waveform in Figure 4.40 using MATLAB.

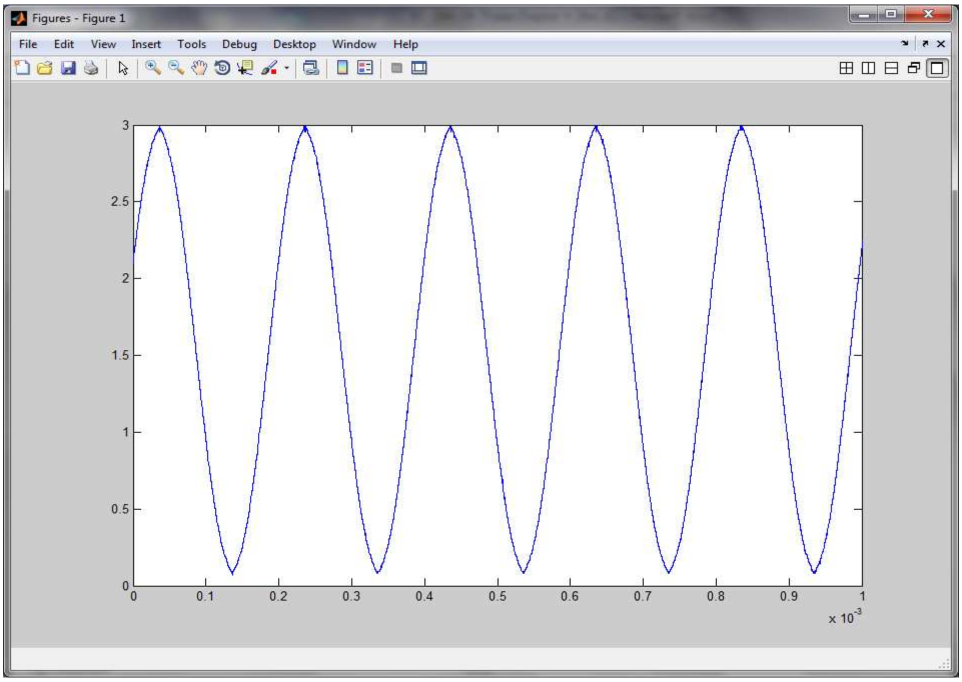


Figure 4.42 The waveform of sinusoidal signal at 5 kHz from a function generator in MATLAB.

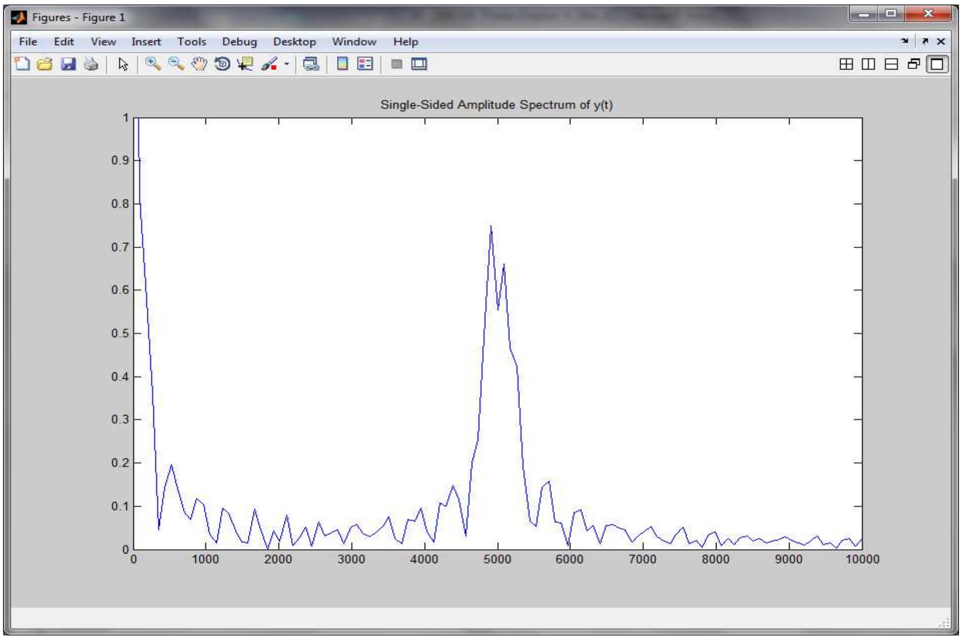


Figure 4.43 The corresponding power spectral density in dB of the waveform in Figure 4.42 using MATLAB.

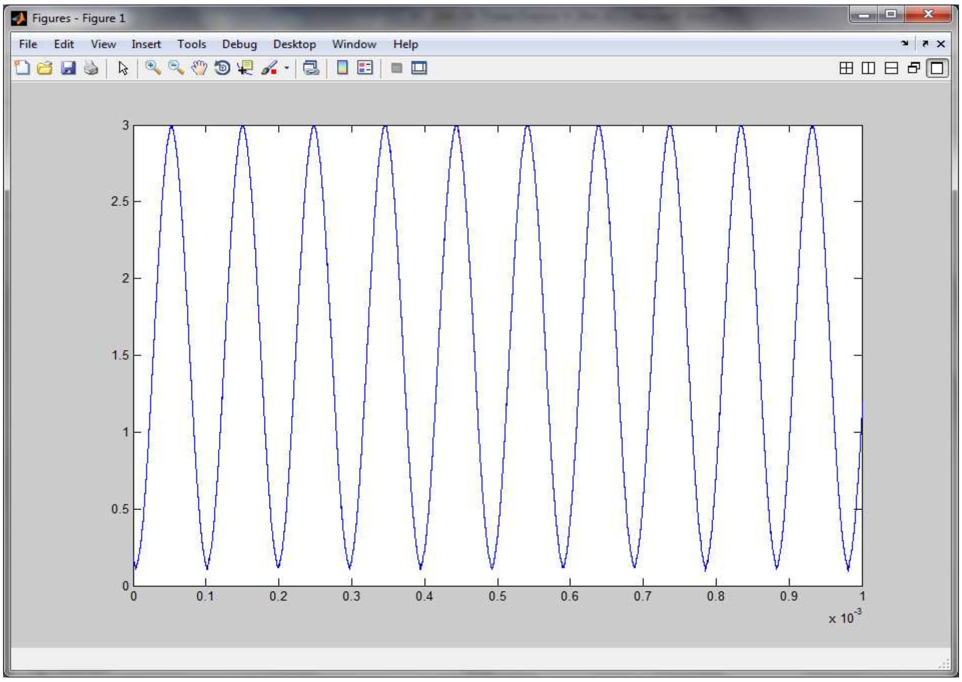


Figure 4.44 The waveform of sinusoidal signal at 10 kHz from a function generator in MATLAB.

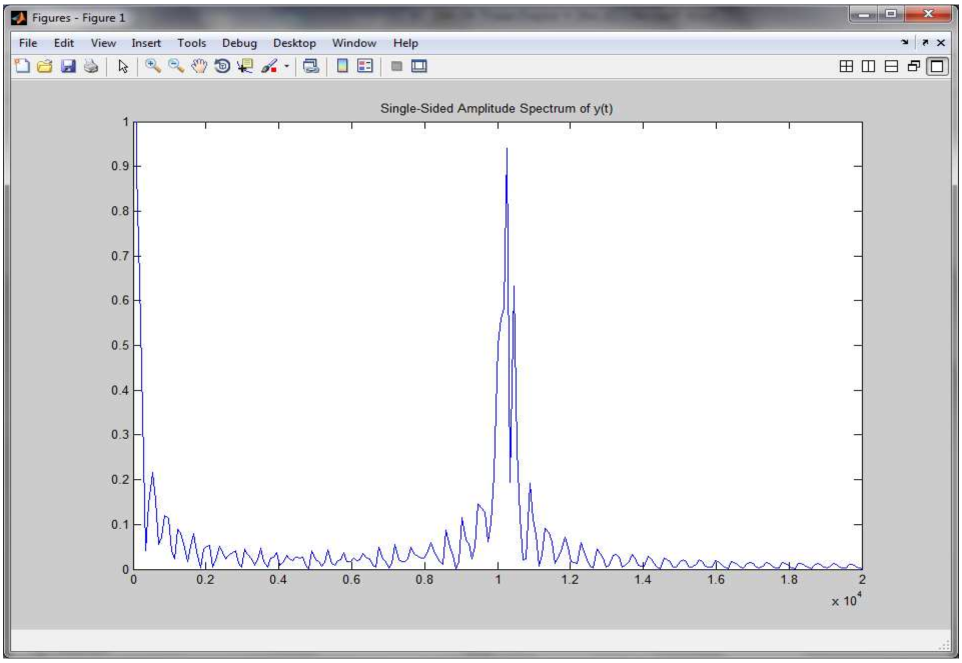


Figure 4.45 The corresponding power spectral density in dB of the waveform in Figure 4.44 using MATLAB.

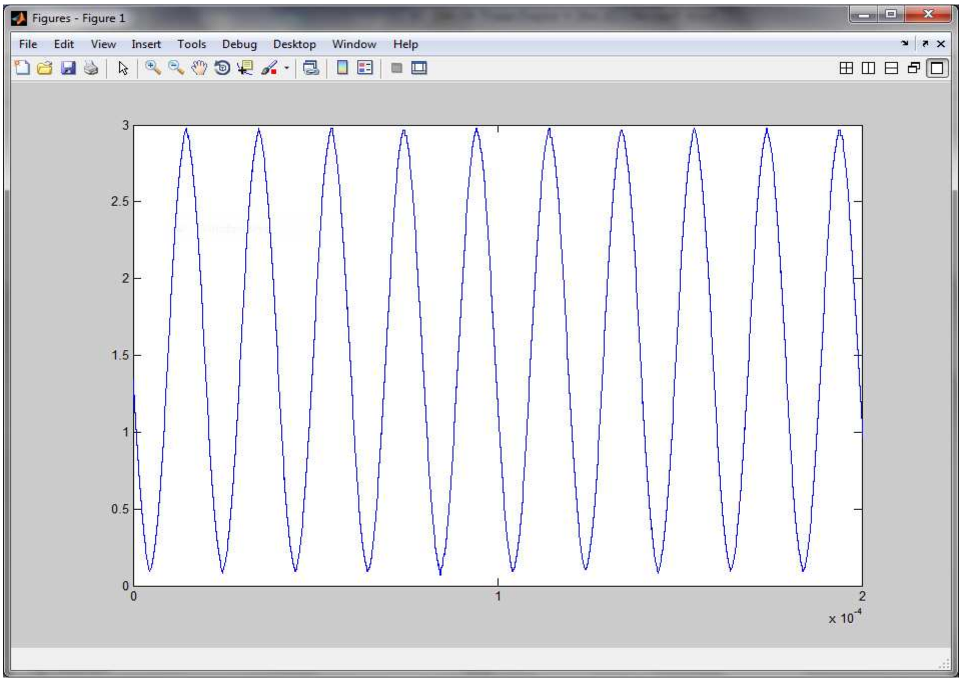


Figure 4.46 The waveform of sinusoidal signal at 50 kHz from a function generator in MATLAB.

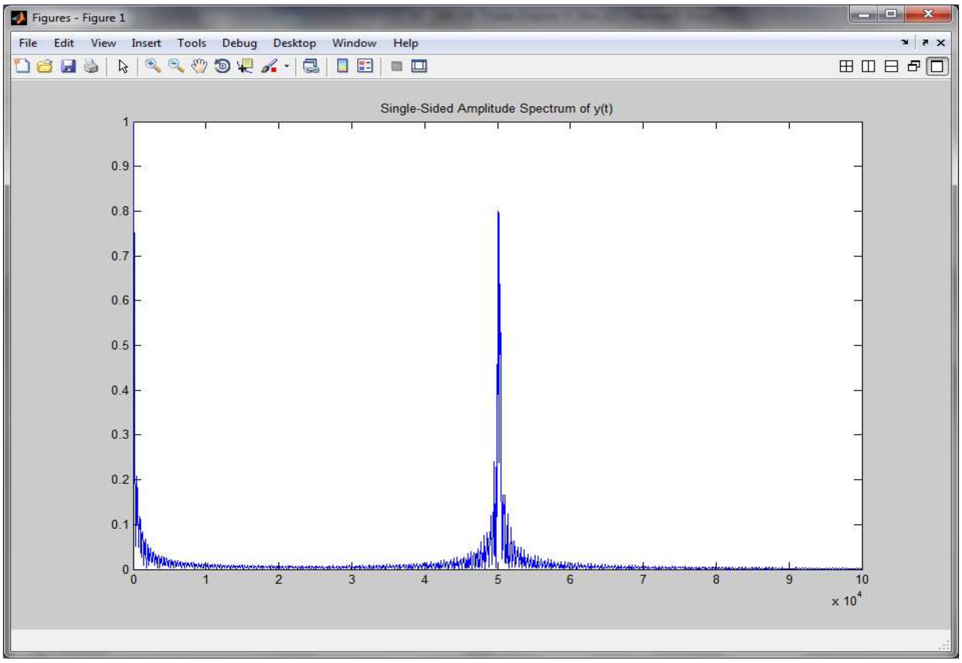


Figure 4.47 The corresponding power spectral density in dB of the waveform in Figure 4.46 using MATLAB.

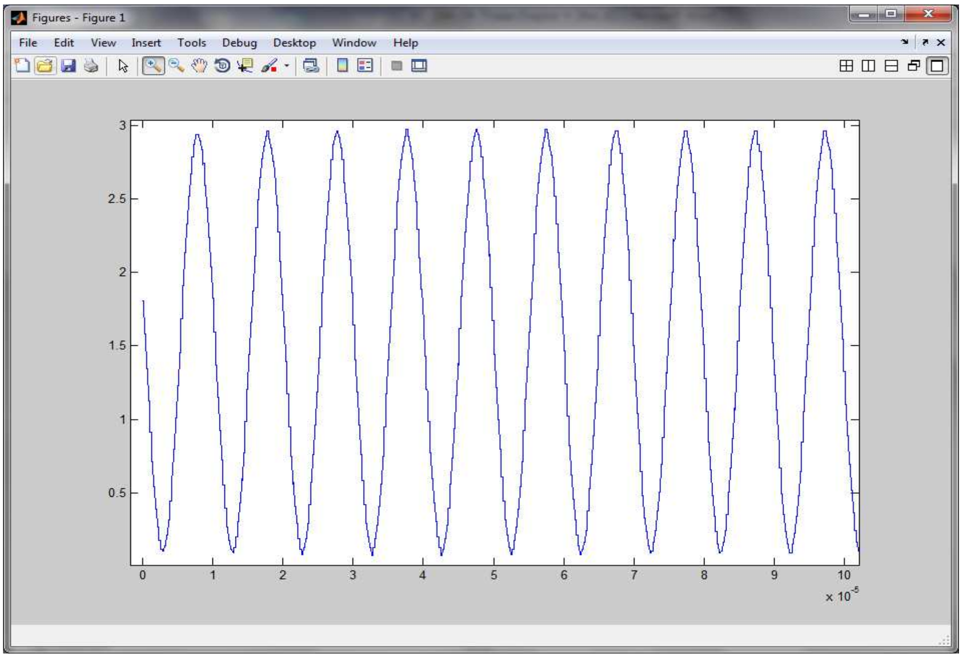


Figure 4.48 The waveform of sinusoidal signal at 100 kHz from a function generator in MATLAB.

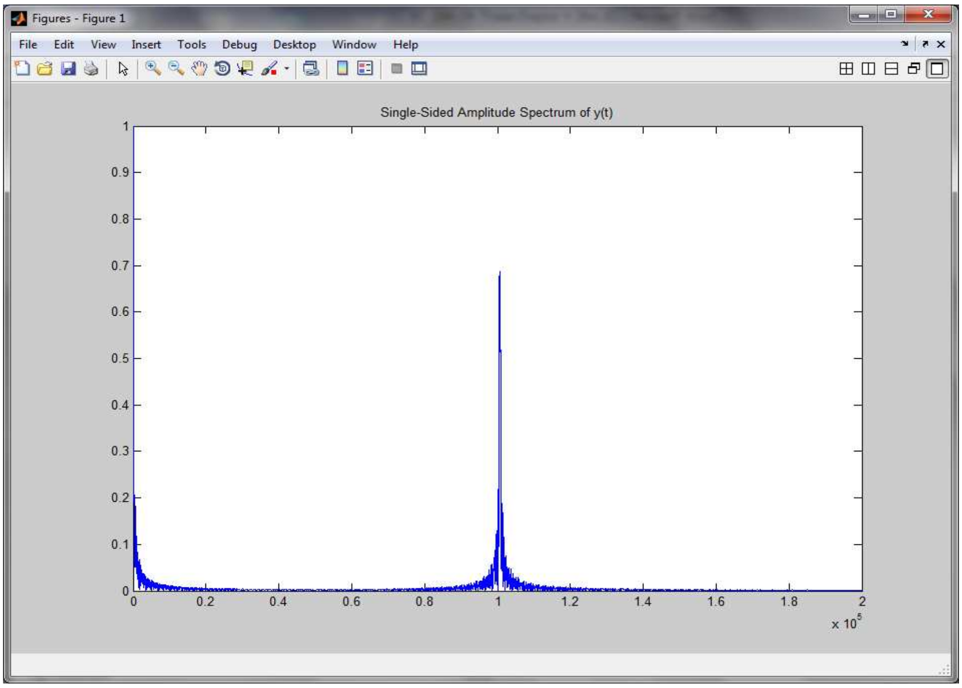


Figure 4.49 The corresponding power spectral density in dB of the waveform in Figure 4.48 using MATLAB.

4.7 AE Standard Lead Break Tests

The commercially available AE sensor is calibrated using the standard pencil lead break tests on an aluminum plate according to the artificial source, i.e. ASTM Standard No. E976-84 [15]. The commercial preamplifier with three selectable different gains at 20dB, 40dB and 60dB were exploits for investigating the magnitudes and the number of counts based on different threshold voltage as shown in Table 4.3. As for instance in the case of the standard AE signal tests and calibrations at 60-dB amplification, Figures 4.50 and 4.51 show the waveforms of the detected AE signals in time domain and the corresponding power spectral density in dB, respectively. Both graphs are obtained from LabVIEW.

Table 4.3 AE count from pencil lead break tests with three selectable different gains at the different distance between AE sensor and the pencil lead.

Gain	Distance	Threshold (V)														
(dB)	(cm)	0.01	0.02	0.03	0.04	0.05	0.10	0.20	0.30	0.40	0.50	1.00	2.00	3.00	4.00	5.00
20	5	21	20	17	13	10	-	-	-	-	-	-	-	-	-	-
	10	-	23	18	8	-	-	-	-	-	-	-	-	-	-	-
40	5	-	-	-	-	-	27	19	12	8	-	-	-	-	-	-
	10	-	-	-	-	-	24	12	10	9	7	-	-	-	-	-
60	5	-	-	-	-	-	-	-	-	-	43	40	26	14	7	-
	10	-	-	-	-	-	-	-	-	-	29	29	10	6	2	-

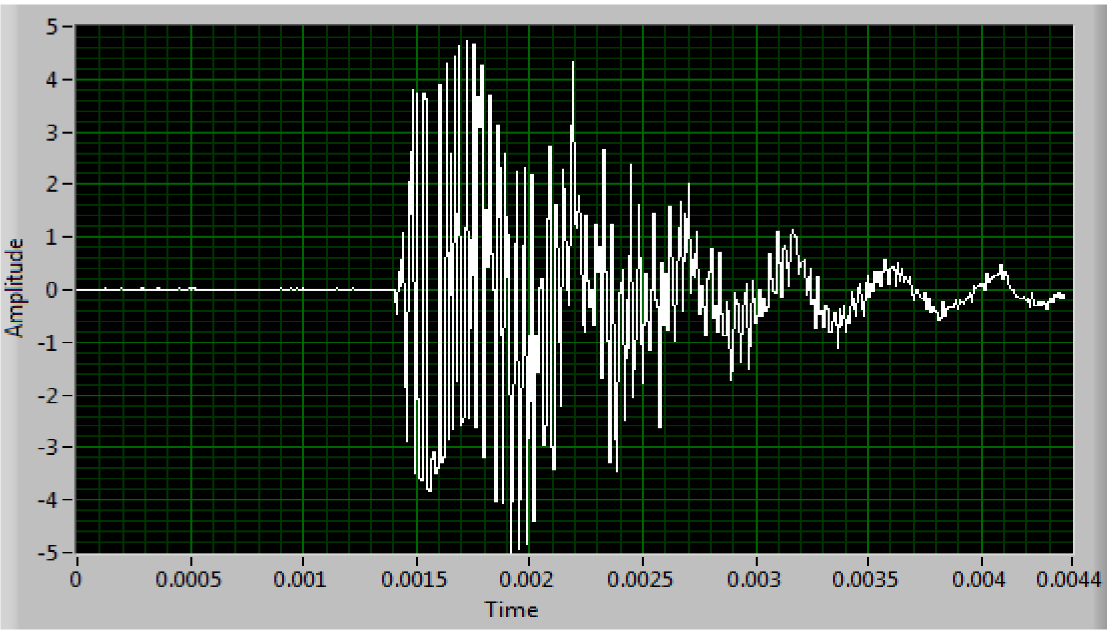


Figure 4.50 The waveforms of the detected AE signals in time domain in the case of the standard AE signal tests and calibrations at 60-dB amplification.

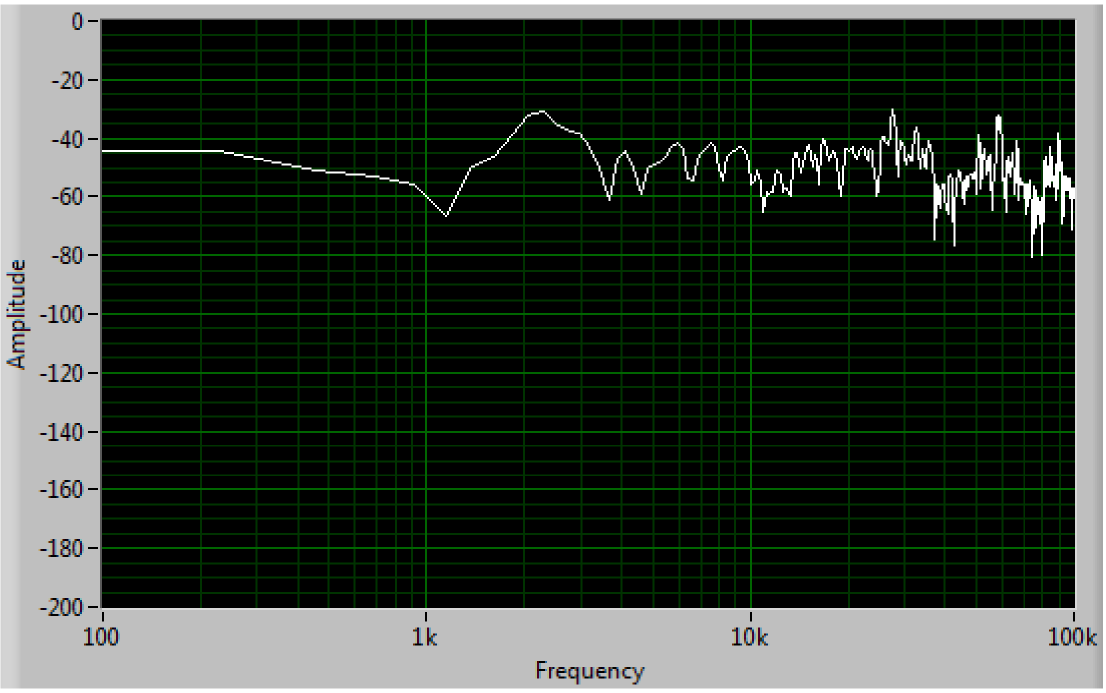


Figure 4.51 The corresponding power spectral density in dB in the case of the standard AE signal tests and calibrations at 60-dB amplification.

4.8 AE Signal Detection using STM32F4

The experimental results have been performed through the protoboard corresponding to the block diagram in Figures 4.52 (a) and (b) as illustrated in Figure 4.53. Figure 4.54 shows the results of the proposed system that detects the AE signals in time domain. It is seen in Figure 4.54 that the use of the proposed preamplifier and STM32F4 microcontroller can also potentially detect the input AE signals.

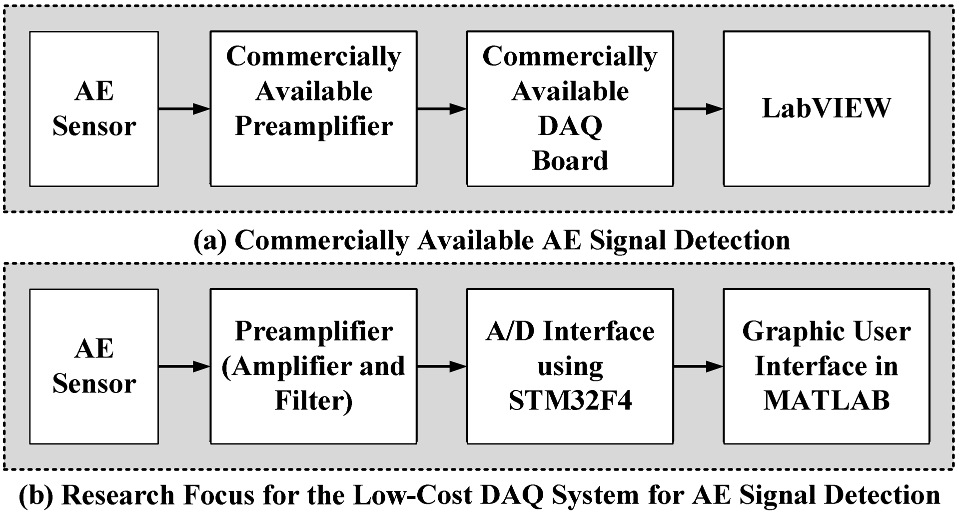


Figure 4.52 General perspective on (a) a typical and (b) the proposed AE signal detection systems using low-cost preamplifier and STM32F4 Microcontroller.

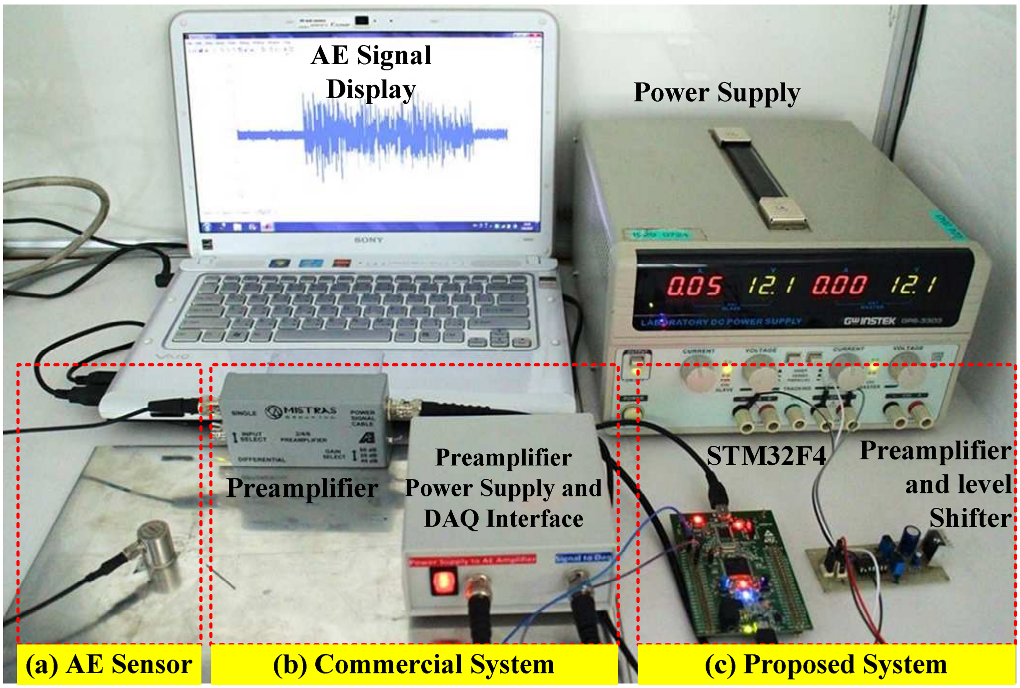


Figure 4.53 The experimental apparatus of (a) a typical and (b) the proposed AE signal detection systems.

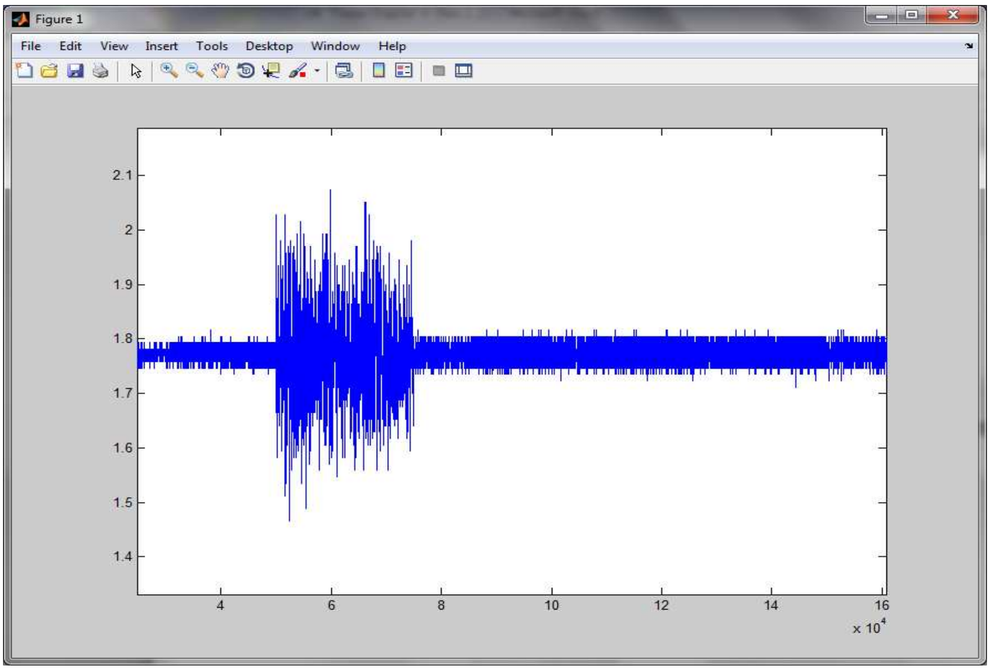


Figure 4.54 The result of the proposed system that detects the AE signals in time domain.

4.9 Overall System Test with the Closed-Loop Water Cooling System of an Induction Heater

After each test are performed, finally, the overall proposed system is applied to a real application at the closed-loop water cooling system of an induction heater in the galvanizing steel plant as shown in Figure 4.55. This induction heater is used for heating of the zinc and steel surface to be an alloy in which very useful for the auto part production. The closed-loop water cooling system is used to keep the operating temperature in normal condition. The existing protection system is depended on water flow rate, water pressure and temperature sensor which are installed in the system as shown in Figures 4.56 and 4.57.

The experimental set up for finding the relationship between AE signals in both time and frequency domains at particular conditions of water pressures and flow rates is performed on the closed-loop water cooling system as shown in Figure 4.58. The AE sensor is attached with stainless steel pipe in which the diameter is 4 inches. The pressure and flow rate are monitored from the pressure gauge and flow meter as shown in Figures 4.56 and 4.57 respectively. Figure 4.59 illustrates the measuring result before attaching the AE sensor to the pipeline using the commercial preamplifier and digital oscilloscope. First, the waveform of AE signal is measured in normal pressure and flow rate at 6.0 bars and 70 litres/minute respectively, which is shown as Figure 4.60. Afterward, pressure is reduced by a manual valve to be 5.5 bar and flow rate of 65 litres/minute, which the waveform of AE signal is changed as shown in figure 4.61. Subsequently, pressure is reduced again to be 5.0 bar and flow rate of 60 litres/minute which the waveform of AE signal is changed as shown in Figure 4.62. The same experiments are performed using the proposed system in which the experimental results are shown as Figures 4.63-4.66. Finally, the experimental set up to transmit AE signal from the AE sensor at the induction heater wirelessly to the personal computer in the control room in which the distance is about 20 m is performed as shown in Figures 4.67 and 4.68. The experimental results show that the proposed monitoring system can detect the turbulent flow in case of the valve position is changed and transmit AE signal wirelessly to the personal computer in the control room.

Figure 4.55 Illustration of the closed-loop water cooling system of an induction heater in the galvanizing steel plant.

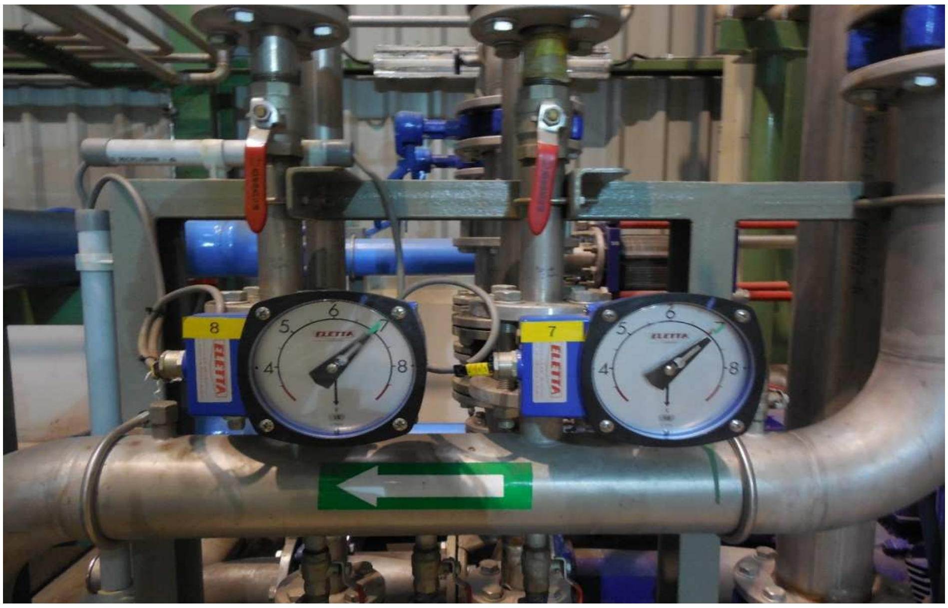


Figure 4.56 Illustration of the flow meters in the closed-loop water cooling system of an induction heater.

Figure 4.57 Illustration of the pressure gauge in the closed-loop water cooling system of an induction heater.

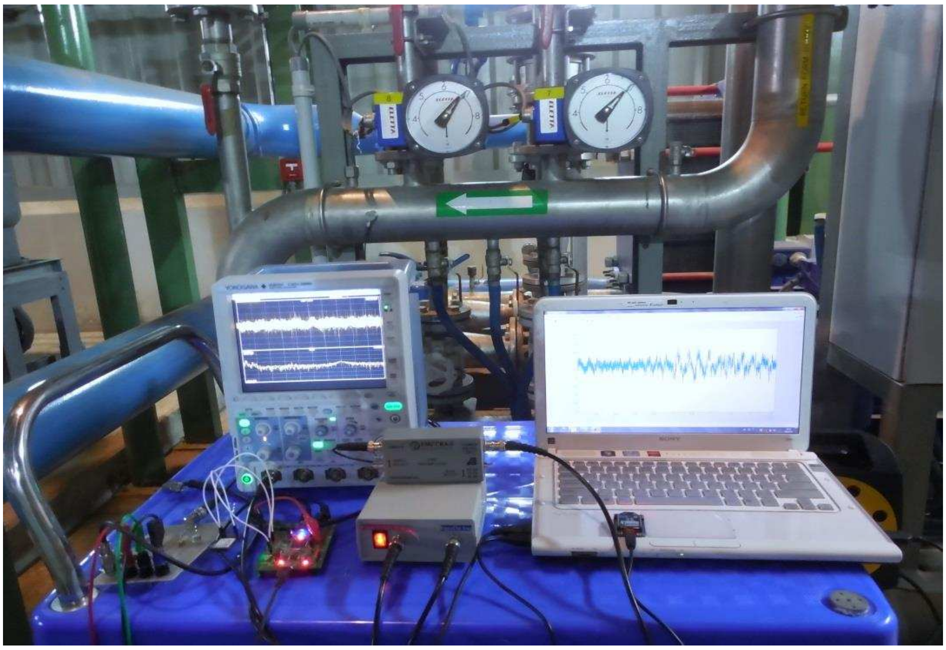


Figure 4.58 The experimental set up for finding the relationship between AE signals at particular conditions of water pressures and flow rates.

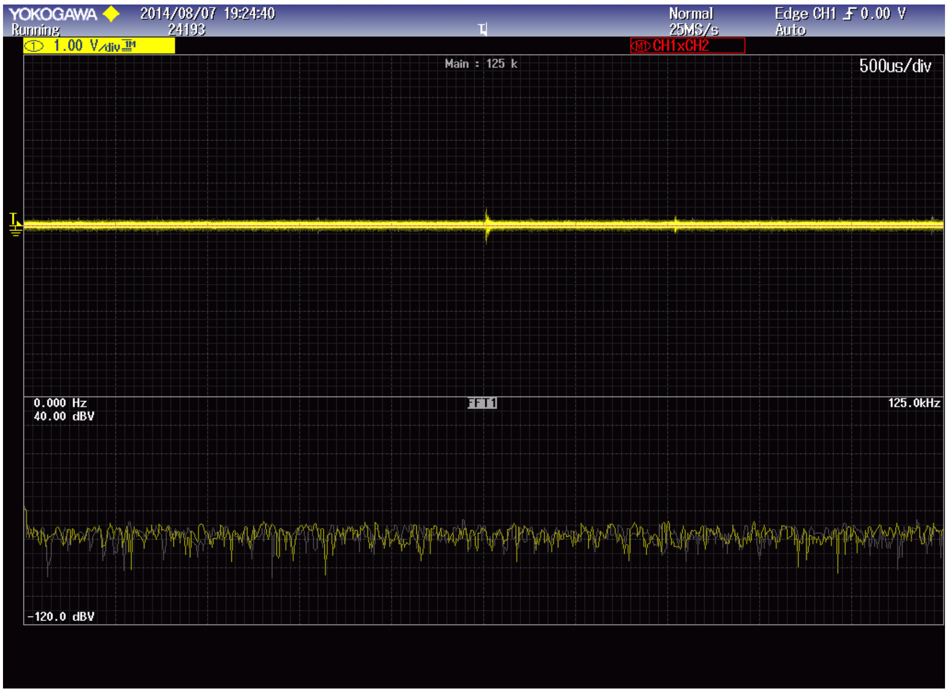


Figure 4.59 The experimental result in case of not connect an AE sensor to the pipeline using the commercial system.

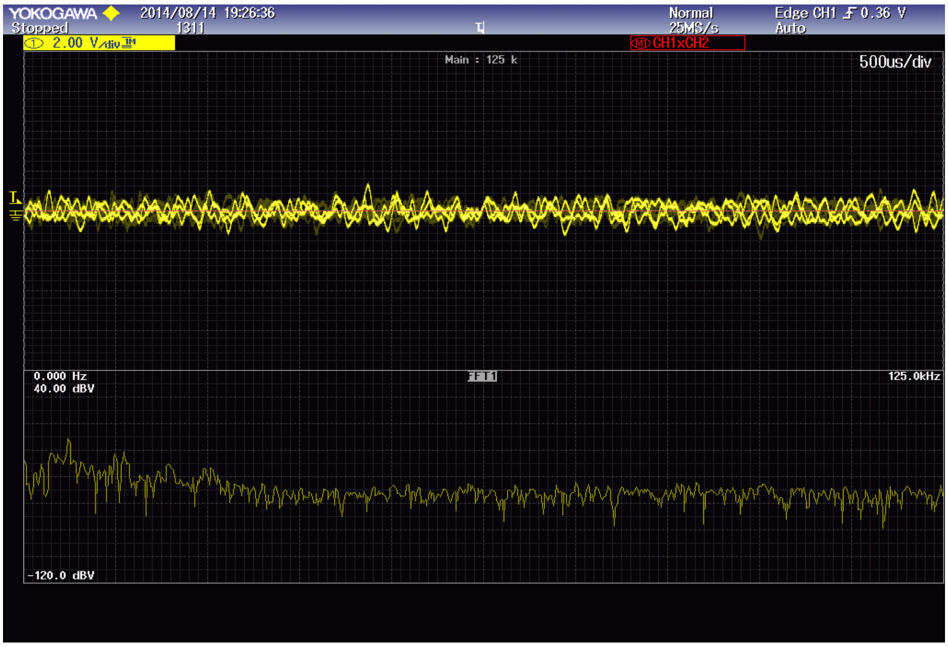


Figure 4.60 The experimental result in case of pressure is 6.0 bars and flow rate is 70 litres/minute using the commercial system.

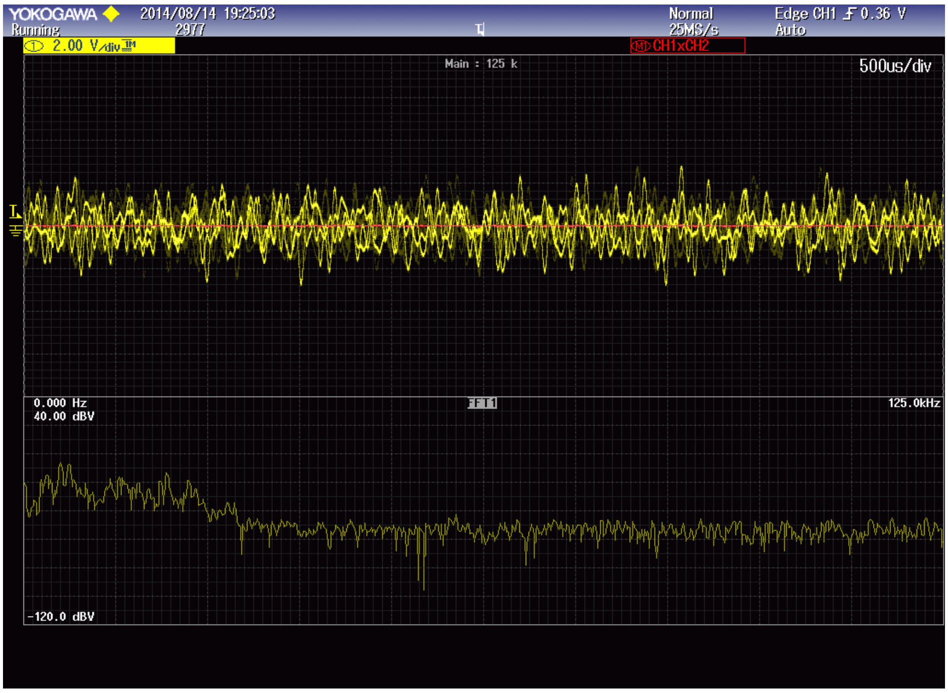


Figure 4.61 The experimental result in case of pressure is 5.5 bars and flow rate is 65 litres/minute using the commercial system.

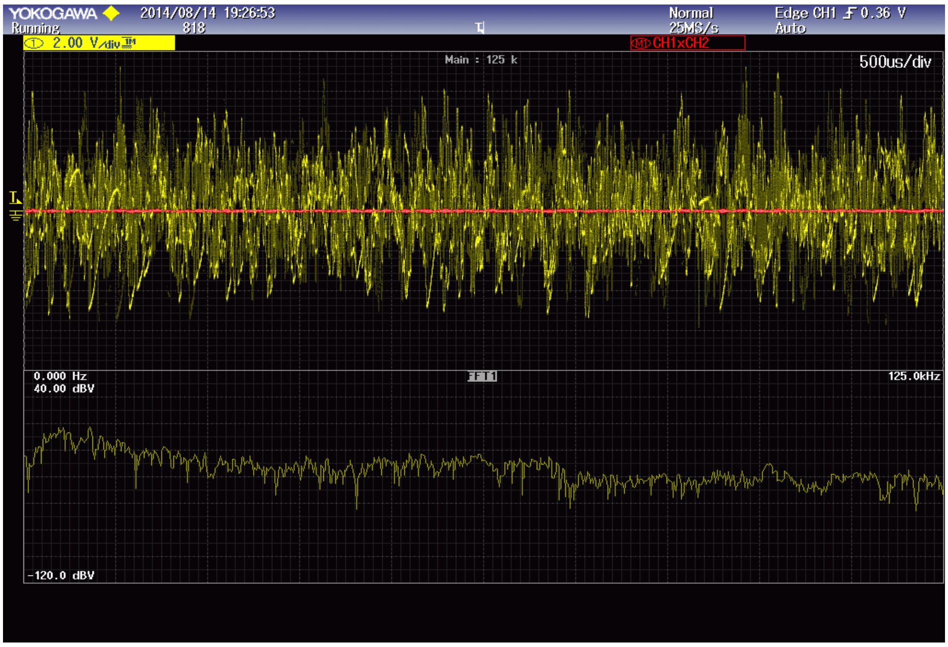


Figure 4.62 The experimental result in case of pressure is 5.0 bars and flow rate is 60 litres/minute using the commercial system.

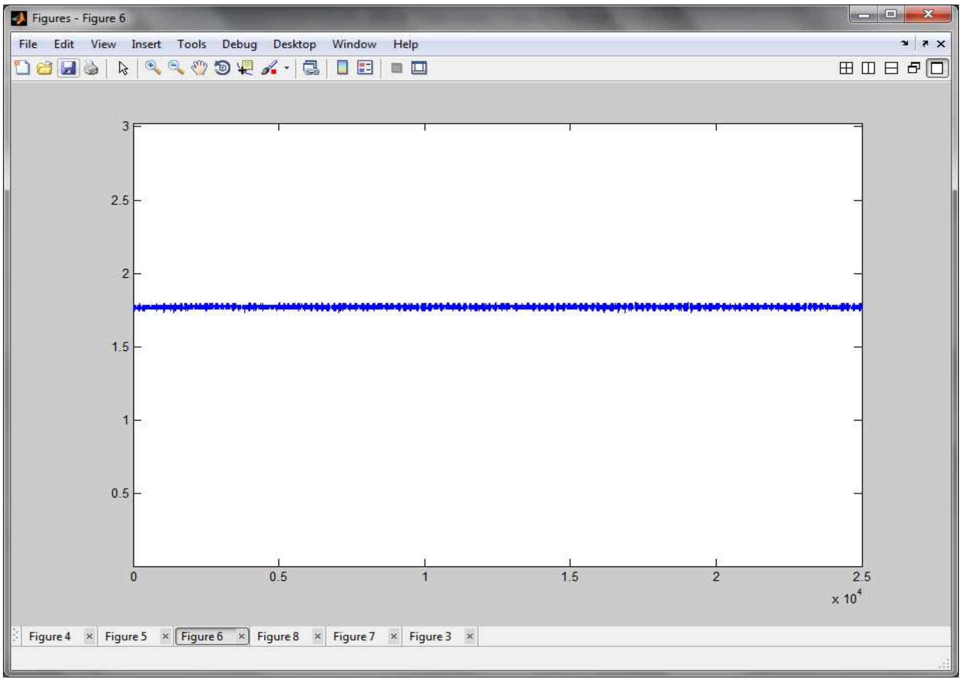


Figure 4.63 The experimental result in case of not connect an AE sensor to the pipeline using the proposed system.

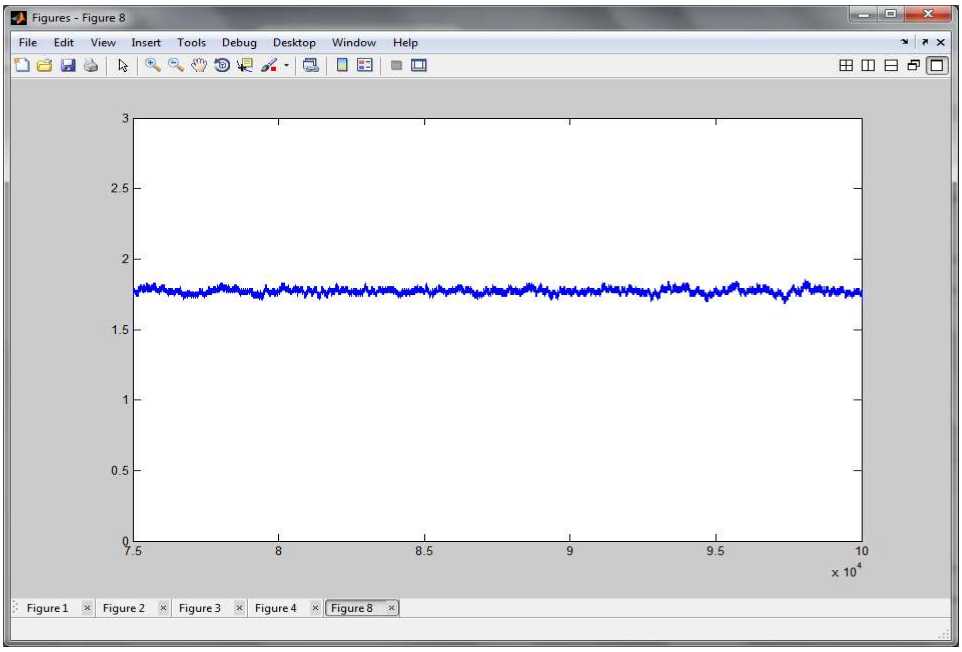


Figure 4.64 The experimental result in case of pressure is 6.0 bars and flow rate is 70 litres/minute using the proposed system.

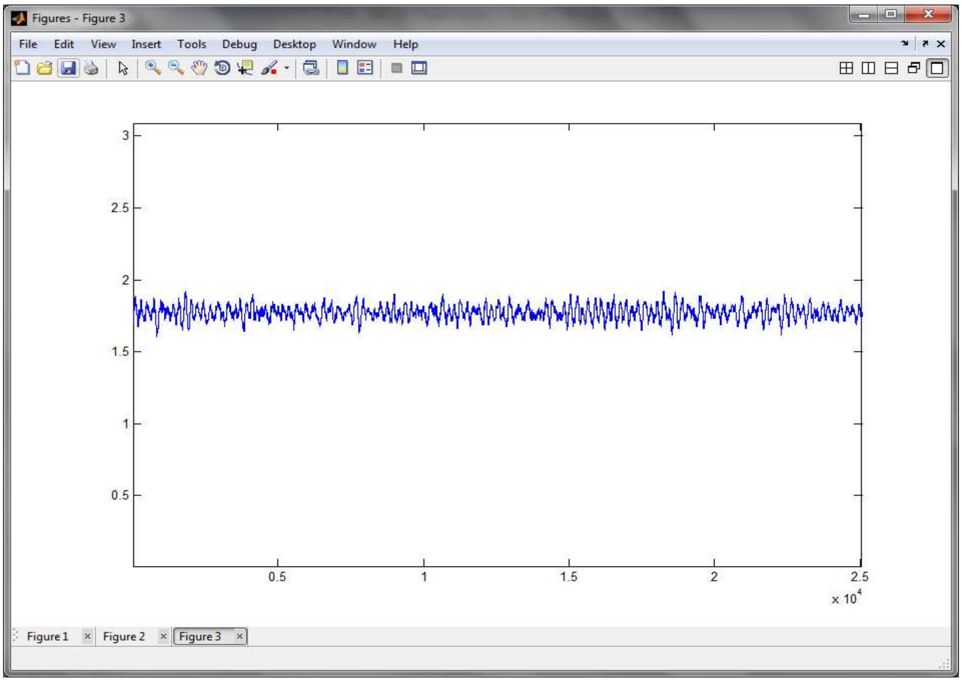


Figure 4.65 The experimental result in case of pressure is 5.5 bars and flow rate is 65 litres/minute using the proposed system.

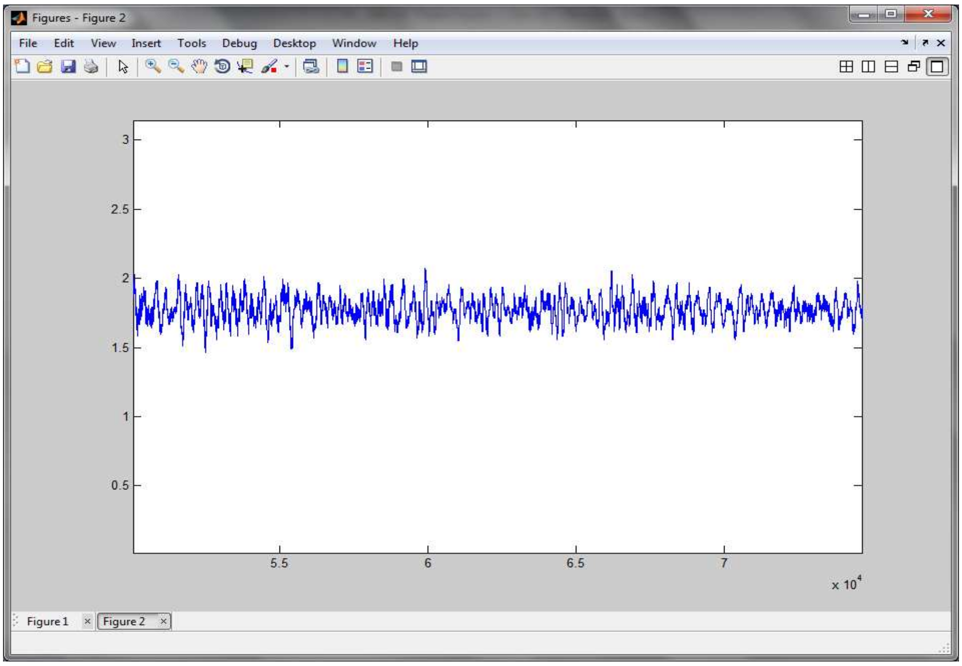


Figure 4.66 The experimental result in case of pressure is 5.0 bars and flow rate is 60 litres/minute using the proposed system.

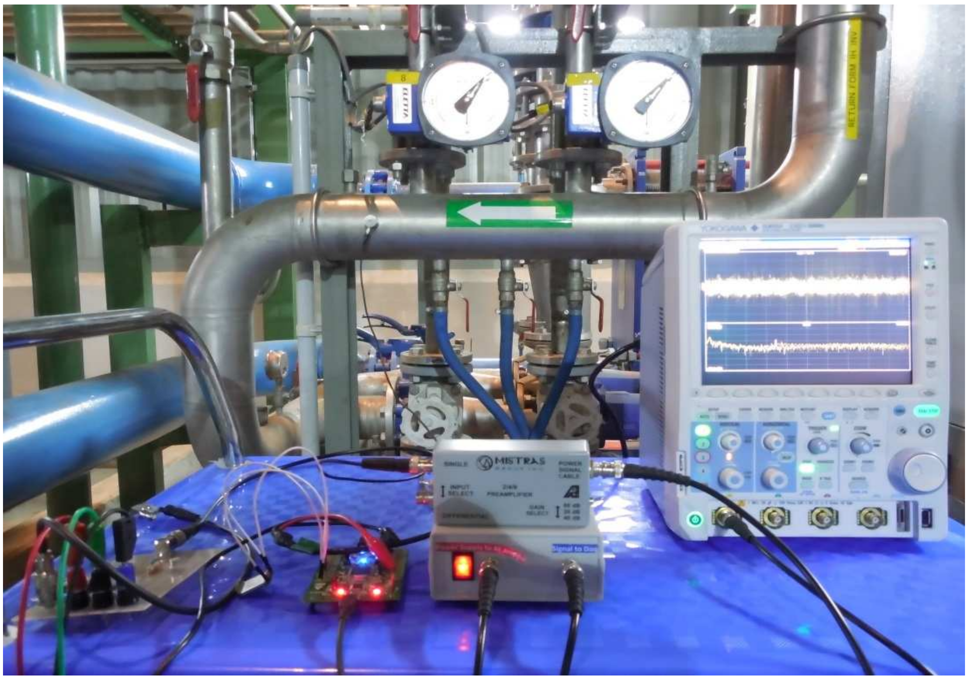


Figure 4.67 The experimental set up to transmit AE signal from the AE sensor at the induction heater wirelessly to personal computer (transceiver side).

Figure 4.68 The experimental set up to transmit AE signal from the AE sensor at the induction heater wirelessly to personal computer (receiver side).

Chapter 5
Conclusion and Suggestion

5.1 Conclusion

In present, the available technology to detect the fault in closed-loop water cooling systems is still complicated, requiring long time for detection as well as high investment cost such as water flow and water pressure transmitters. This research has aimed to apply the AE sensor for a real-time monitoring system of closed-loop water cooling system by detecting the AE signal generated by turbulent flows of cooling water. The proposed system comprises five major components, i.e. a preamplifier, a STM32F4 Microcontroller, a wireless data transceiver, a wireless data receiver and a graphic user interface (GUI). The preamplifier is implemented using BiMOS operational amplifier model CA3140 which provides relatively high input impedance and high-speed performance. It is obviously seen through the results that the quality signal analysis were strongly satisfy including high frequency response at high voltage gain, obtain the peak of signal at 100 kHz. The generated signal was successfully digitized by STM32F4 Microcontroller and transmits to a MATLAB via wireless data transmission. In addition, the proposed system offers a potential alternative cost-effective data acquisition to commercially available data acquisition systems in the same range of frequency response such as National Instrument (NI), Physical Acoustics Corporation (PAC) etc. in which the cost could be reduced to approximately 80% comparing to the specified DAQs. The overall system is applicable for a real-time monitoring system of closed-loop water cooling system. The tests have been performed with the closed-loop water cooling system of the induction heater in the galvanizing steel plant in which the experimental results show that the proposed monitoring system can detect the turbulent flow, which is occurred in case of the valve position is changed.

5.2 Suggestion

As the proposed system in this thesis is still in an design phase for the application of the AE sensor for a real-time monitoring system of closed-loop water cooling system, in order to increase the reliability and benefit, the system are needed to improve as follows;

5.2.1 The AE signal transceiver should be improved as a stand-alone device including an integrated power supply and permanent connectors in order to increase flexibility to use in the real world application.

5.2.2 The GUI should be improved by addition of the fault classification function with alarm messages using the ANN data analysis such as fault clustering.

5.2.3 The proposed system should be developed using the multi AE sensors network in order to increase accuracy and capability for utilization in industries.

5.2.4 The proposed system should be developed by addition of digital output function to send alarm signal or machine stop command to the related machine in order to prevent major troubles.

Reference

References

[1] NDT Resource Center. (2011). **Introduction to Acoustic Emission Testing.** Retrieved August 1, 2014, from https://www.nde-ed.org/ Education Resources/CommunityCollege/Other Methods/AE/AE_Intro.htm.

[2] C. Grosse; and M. Ohtsu, Eds. (2008). **Acoustic Emission Testing: Basics for Research, Applications in Civil Engineering.** Springer Berlin Heidelberg. P. 8, 10, 11, 16.

[3] **Standard Terminology for Nondestructive Examinations**. (n.d.). ASTM Standard No. E1316.

[4] Majeed MA; and Murthy CRL. (2001). A Model with Nonzero Rise Time for AE Signal. **Sadhana** . 2001 (25-5) : 465-474.

[5] Jiao Yang; Yang Qingxin; Li Guanghai; and Zhang Jingyan. (2006). Acoustic Emission Source Identification Technique for Buried Gas Pipeline Leak**. IEEE.** ICARCV2006. P.1-5.

[6] Athanasios Anastasopoulos; Dimitrios Kourousis; and Konstantinos Bolas. (2009). Acoustic Emission Leak Detection of Liquid Filled Buried Pipeline. **J. Acoustic Emission**. 27 (2009) : 27-39.

[7] Zhang Jian-Li; and Guo Wei-Xing. (2011). Study on the Characteristics of the Leakage Acoustic Emission in Cast Iron Pipe by Experiment**. IEEE.** IMCCC2011. P.122-125.

[8] Didem Ozevin; and James Harding. (2012). Novel Leak Localization in Pressurized Pipeline Networks using Acoustic Emission and Geometric Connectivity. **International Journal of Pressure Vessels and Piping**. 92 (2012) : 63-69.

[9] M. Noipitak; A. Prateepasen; and W. Kaewwaewnoi. (2011). A Relative Calibration Method for Valve Leakage Rate Measurement System. **Elsevier Measurement**. 44 (2011) : 211-218.

[10] A. Prateepasen; W. Kaewwaewnoi; and P. Kaewtrakulpong. (2011). Smart Portable Noninvasive Instrument for Detection of Internal Air Leakage of a Valve using Acoustic Emission Signals. **Elsevier Measurement**. 44 (2011) : 378-384.

[11] N.C. Hii; C.K. Tan; S.J. Wilcox; and Z.S. Chong. (2013). An Investigation of the Generation of Acoustic Emission from the Flow of Particulate Solids in Pipelines. **Elsevier Powder Technology**. 243 (2013) : 120-129.

[12] A. Mostafapour; and S. Davoudi. (2013). Analysis of Leakage in High Pressure Pipe using Acoustic Emission Method. **Elsevier Applied Acoustics**. 74 (2013) : 335-342.

[13] Eric Bechhoefer; Yongzhi Qu; Junda Zhu; and David He. (2013). Signal Processing Techniques to Improve an acoustic Emissions Sensor. **Annual Conference of Prognostics of Prognostics and Health Management Society.** 2013 : 1-8.

[14] Zhang Haifeng; Li Zhenlin; Ji Zhongli; Li Hongxing; and Li Mingxiao. (2013). Application of Acoustic Emission and Support Vector Machine to Detect the Leakage of Pipeline Valve**. Conference on Measuring Technology and Mechatronics Automation**. 2013 : 283-286.

[15] **Standard Guide for Detecting the Reproducibility of Acoustic Emission Sensor Response**. (n.d.). ASTM Standard No. E976-84.

Appendix

Appendix A - MATLAB XBee and GUI interface code


```
function output = STM32F4DiscoveryVCP (port)
    dataLength      = 25000;
    inputBufferSize = 10 * dataLength;
    vcp = serial(port);
    set(vcp, 'BaudRate',        1050000);
    set(vcp, 'FlowControl',     'none');
    set(vcp, 'Parity',          'none');
    set(vcp, 'DataBits',        8);
    set(vcp, 'StopBit',         1);
    set(vcp, 'InputBufferSize', inputBufferSize);
    set(vcp, 'ReadAsyncMode',   'continuous');
    set(vcp, 'Timeout',         100);

    fopen(vcp);
    fwrite(vcp, '@');
    rawData = fread(vcp, dataLength);
    rawData = (3.0 / 256) .* rawData;
    fclose(vcp);
    delete(vcp);
    output = rawData;
end
```

```
clear all
clc
port = 'COM3';
hLine = plot(nan);
loop = 15;
len = 25000;
axis([0 (loop * 25000) 0 3])
while 1
   for i = 1: loop
      data(((i - 1) * len) + 1: (i * len)) = STM32F4DiscoveryVCP(port);
   end
   found = find(data >= 1.6, 1);
   if ~isempty(found)
      set(hLine, 'YData', data(found:end));
      drawnow
      break
   end
end
```

Appendix B – STM32F4 – A/D and Data Transmission code

stm32f4xx_conf.h

```
#ifndef __STM32F4xx_CONF_H
#define __STM32F4xx_CONF_H
#include "stm32f4xx_adc.h"
#include "stm32f4xx_can.h"
#include "stm32f4xx_crc.h"
#include "stm32f4xx_cryp.h"
#include "stm32f4xx_dac.h"
#include "stm32f4xx_dbgmcu.h"
#include "stm32f4xx_dcmi.h"
#include "stm32f4xx_dma.h"
#include "stm32f4xx_exti.h"
#include "stm32f4xx_flash.h"
#include "stm32f4xx_fsmc.h"
#include "stm32f4xx_hash.h"
#include "stm32f4xx_gpio.h"
#include "stm32f4xx_i2c.h"
#include "stm32f4xx_iwdg.h"
#include "stm32f4xx_pwr.h"
#include "stm32f4xx_rcc.h"
#include "stm32f4xx_rng.h"
#include "stm32f4xx_rtc.h"
#include "stm32f4xx_sdio.h"
```

```
#include "stm32f4xx_spi.h"

#include "stm32f4xx_syscfg.h"

#include "stm32f4xx_tim.h"

#include "stm32f4xx_usart.h"

#include "stm32f4xx_wwdg.h"

#include "misc.h"

#ifdef  USE_FULL_ASSERT

#define assert_param(expr) ((expr) ? (void)0 : assert_failed((uint8_t *)__FILE__,
__LINE__))

  void assert_failed(uint8_t* file, uint32_t line);

#else

#define assert_param(expr) ((void)0)

#endif

#endif
```

system_stm32f4xx.c

```
#include "stm32f4xx.h"

#define VECT_TAB_OFFSET  0x00

#define PLL_M      8

#define PLL_N      336

#define PLL_P      2

#define PLL_Q      7

  uint32_t SystemCoreClock = 168000000;

  __I uint8_t AHBPrescTable[16] = {0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 3, 4, 6, 7, 8, 9};
```

```
static void SetSysClock(void);
#ifdef DATA_IN_ExtSRAM
  static void SystemInit_ExtMemCtl(void);
#endif /* DATA_IN_ExtSRAM */
void SystemInit(void)
{
  #if (__FPU_PRESENT == 1) && (__FPU_USED == 1)
        SCB->CPACR |= ((3UL << 10*2)|(3UL << 11*2));
  #endif
  RCC->CR |= (uint32_t)0x00000001;
  RCC->CFGR = 0x00000000;
  RCC->CR &= (uint32_t)0xFEF6FFFF;
  RCC->PLLCFGR = 0x24003010;
  RCC->CR &= (uint32_t)0xFFFBFFFF;
  RCC->CIR = 0x00000000;
#ifdef DATA_IN_ExtSRAM
  SystemInit_ExtMemCtl();
#endif
  SetSysClock();
#ifdef VECT_TAB_SRAM
#else
#endif
}
void SystemCoreClockUpdate(void)
```

```
{
 uint32_t tmp = 0, pllvco = 0, pllp = 2, pllsource = 0, pllm = 2;
 tmp = RCC->CFGR & RCC_CFGR_SWS;
 switch (tmp)
 {
  case 0x00:
   SystemCoreClock = HSI_VALUE;
   break;
  case 0x04:
   SystemCoreClock = HSE_VALUE;
   break;
  case 0x08:
   pllsource = (RCC->PLLCFGR & RCC_PLLCFGR_PLLSRC) >> 22;
   pllm = RCC->PLLCFGR & RCC_PLLCFGR_PLLM;

   if (pllsource != 0)
   {
    pllvco = (HSE_VALUE / pllm) * ((RCC->PLLCFGR &
    RCC_PLLCFGR_PLLN) >> 6);
   }
   else
   {
    pllvco = (HSI_VALUE / pllm) * ((RCC->PLLCFGR &
    RCC_PLLCFGR_PLLN) >> 6);
```

```
    }
    pllp = (((RCC->PLLCFGR & RCC_PLLCFGR_PLLP) >>16) + 1 ) *2;
    SystemCoreClock = pllvco/pllp;
    break;
    default:
    SystemCoreClock = HSI_VALUE;
    break;
  }
  tmp = AHBPrescTable[((RCC->CFGR & RCC_CFGR_HPRE) >> 4)];
  SystemCoreClock >>= tmp;
}
static void SetSysClock(void)
{
  __IO uint32_t StartUpCounter = 0, HSEStatus = 0;
  RCC->CR |= ((uint32_t)RCC_CR_HSEON);
  do
  {
    HSEStatus = RCC->CR & RCC_CR_HSERDY;
    StartUpCounter++;
  } while((HSEStatus == 0) && (StartUpCounter != HSE_STARTUP_TIMEOUT));

  if ((RCC->CR & RCC_CR_HSERDY) != RESET)
  {
    HSEStatus = (uint32_t)0x01;
```

```
}
else
{
  HSEStatus = (uint32_t)0x00;
}
if (HSEStatus == (uint32_t)0x01)
{
  RCC->APB1ENR |= RCC_APB1ENR_PWREN;
  PWR->CR |= PWR_CR_PMODE;
  RCC->CFGR |= RCC_CFGR_HPRE_DIV1;
  RCC->CFGR |= RCC_CFGR_PPRE2_DIV2;
  RCC->CFGR |= RCC_CFGR_PPRE1_DIV4;
  RCC->PLLCFGR = PLL_M | (PLL_N << 6) | (((PLL_P >> 1) -1) << 16) |
                 (RCC_PLLCFGR_PLLSRC_HSE) | (PLL_Q << 24);
  RCC->CR |= RCC_CR_PLLON;
  while((RCC->CR & RCC_CR_PLLRDY) == 0)
  {
  }
  FLASH->ACR = FLASH_ACR_ICEN |FLASH_ACR_DCEN
  |FLASH_ACR_LATENCY_5WS;
  RCC->CFGR &= (uint32_t)((uint32_t)~(RCC_CFGR_SW));
  RCC->CFGR |= RCC_CFGR_SW_PLL;
  while ((RCC->CFGR & (uint32_t)RCC_CFGR_SWS ) !=
  RCC_CFGR_SWS_PLL);
  {
```

```
    }
  }
  else
  {
  }
}
#ifdef DATA_IN_ExtSRAM
void SystemInit_ExtMemCtl(void)
{
  RCC->AHB1ENR   = 0x00000078;
  GPIOD->AFR[0]  = 0x00cc00cc;
  GPIOD->AFR[1]  = 0xcc0ccccc;
  GPIOD->MODER   = 0xaaaa0a0a;
  GPIOD->OSPEEDR = 0xffff0f0f;
  GPIOD->OTYPER  = 0x00000000;
  GPIOD->PUPDR   = 0x00000000;
  GPIOE->AFR[0]  = 0xc00cc0cc;
  GPIOE->AFR[1]  = 0xcccccccc;
  GPIOE->MODER   = 0xaaaa828a;
  GPIOE->OSPEEDR = 0xffffc3cf;
  GPIOE->OTYPER  = 0x00000000;
  GPIOE->PUPDR   = 0x00000000;
  GPIOF->AFR[0]  = 0x00cccccc;
  GPIOF->AFR[1]  = 0xcccc0000;
```

```
  GPIOF->MODER   = 0xaa000aaa;
  GPIOF->OSPEEDR = 0xff000fff;
  GPIOF->OTYPER  = 0x00000000;
  GPIOF->PUPDR   = 0x00000000;
  GPIOG->AFR[0]  = 0x00cccccc;
  GPIOG->AFR[1]  = 0x000000c0;
  GPIOG->MODER   = 0x00080aaa;
  GPIOG->OSPEEDR = 0x000c0fff;
  GPIOG->OTYPER  = 0x00000000;
  GPIOG->PUPDR   = 0x00000000;
  RCC->AHB3ENR   = 0x00000001;
  FSMC_Bank1->BTCR[2]  = 0x00001015;
  FSMC_Bank1->BTCR[3]  = 0x00010603;//0x00010400;
  FSMC_Bank1E->BWTR[2] = 0x0fffffff;
}
#endif
```

stm32f4xx_it.h

```
#ifndef __STM32F4xx_IT_H
#define __STM32F4xx_IT_H
#ifdef __cplusplus
 extern "C" {
#endif
#include "stm32f4xx.h"
```

```
void NMI_Handler(void);
void HardFault_Handler(void);
void MemManage_Handler(void);
void BusFault_Handler(void);
void UsageFault_Handler(void);
void SVC_Handler(void);
void DebugMon_Handler(void);
void PendSV_Handler(void);
void SysTick_Handler(void);
void DMA2_Stream0_IRQHandler(void);
#ifdef __cplusplus
}
#endif
#endif
```

stm32f4xx_it.c

```
#include "stm32f4xx_it.h"
#include "usb_core.h"
#include "usbd_core.h"
#include "stm32f4_discovery.h"
#include "usbd_cdc_core.h"
#include "usb_cdc_cmd.h"
extern USB_OTG_CORE_HANDLE          USB_OTG_dev;
extern uint32_t USBD_OTG_ISR_Handler (USB_OTG_CORE_HANDLE *pdev);
```

```
void NMI_Handler(void) {
}
void HardFault_Handler(void) {
  while (1);
}
void MemManage_Handler(void) {
  while (1);
}
void BusFault_Handler(void) {
  while (1);
}
void UsageFault_Handler(void) {
  while (1);
}
void SVC_Handler(void) {
}
void DebugMon_Handler(void) {
}
void PendSV_Handler(void) {
}
void SysTick_Handler(void) {
}
#ifdef USE_USB_OTG_FS
void OTG_FS_WKUP_IRQHandler(void) {
```

```
 if (USB_OTG_dev.cfg.low_power) {
    *(uint32_t *)(0xE000ED10) &= 0xFFFFFFF9 ;
    SystemInit();
    USB_OTG_UngateClock(&USB_OTG_dev);
  }
  EXTI_ClearITPendingBit(EXTI_Line18);
}
#endif
#ifdef USE_USB_OTG_HS
void OTG_HS_WKUP_IRQHandler(void) {
  if (USB_OTG_dev.cfg.low_power) {
    *(uint32_t *)(0xE000ED10) &= 0xFFFFFFF9 ;
    SystemInit();
    USB_OTG_UngateClock(&USB_OTG_dev);
  }
  EXTI_ClearITPendingBit(EXTI_Line20);
}
#endif
#ifdef USE_USB_OTG_HS
void OTG_HS_IRQHandler(void) {
#else
void OTG_FS_IRQHandler(void) {
#endif
  USBD_OTG_ISR_Handler (&USB_OTG_dev);
```

```
}
#ifdef USB_OTG_HS_DEDICATED_EP1_ENABLED
void OTG_HS_EP1_IN_IRQHandler(void) {
  USBD_OTG_EP1IN_ISR_Handler (&USB_OTG_dev);
}
void OTG_HS_EP1_OUT_IRQHandler(void) {
  USBD_OTG_EP1OUT_ISR_Handler (&USB_OTG_dev);
}
#endif
void DMA2_Stream0_IRQHandler (void) {
  cmd_exec_stop();
  DMA_ClearITPendingBit(DMA2_Stream0, DMA_IT_TCIF0);
}
```

usb_cdc_cmd.h

```
#ifndef __USB_CDC_CMD_H
#define __USB_CDC_CMD_H
#ifdef __cplusplus
extern "C" {
#endif
#include "stm32f4xx.h"
void cmd_exec_function (void);
void cmd_exec_start (void);
void cmd_exec_stop (void);
```

```
#ifdef __cplusplus
}
#endif
#endif
```

usb_cdc_cmd.c

```
#include "usb_cdc_cmd.h"
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "stm32f4xx.h"
#include "stm32f4_discovery.h"
#include "usbd_cdc_core.h"
#include "usb_bsp.h"
#include "usbd_cdc.h"
#include "usbd_usr.h"
#include "usbd_desc.h"
#define ADC_CDR_ADDRESS    ((uint32_t)0x40012308)
__ALIGN_BEGIN USB_OTG_CORE_HANDLE USB_OTG_dev __ALIGN_END;
extern uint8_t  APP_Rx_Buffer [];
extern uint32_t APP_Rx_ptr_in;
extern uint32_t APP_Rx_ptr_out;
uint8_t is_adc_on;
void cmd_exec_function (void) {
```

```
unsigned int Timeout;
ADC_InitTypeDef ADC_InitStructure;
ADC_CommonInitTypeDef ADC_CommonInitStructure;
DMA_InitTypeDef DMA_InitStructure;
GPIO_InitTypeDef GPIO_InitStructure;
NVIC_InitTypeDef NVIC_InitStructure;
USBD_Init(&USB_OTG_dev, USB_OTG_FS_CORE_ID, &USR_desc,
&USBD_CDC_cb, &USR_cb);
RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_DMA2 |
RCC_AHB1Periph_GPIOC, ENABLE);
RCC_APB2PeriphClockCmd(RCC_APB2Periph_ADC1 |
RCC_APB2Periph_ADC2 |
RCC_APB2Periph_ADC3, ENABLE);
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_2;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AN;
GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL;
GPIO_Init(GPIOC, &GPIO_InitStructure);
DMA_DeInit(DMA2_Stream0);
DMA_InitStructure.DMA_Channel            = DMA_Channel_0;
DMA_InitStructure.DMA_PeripheralBaseAddr =
(uint32_t)ADC_CDR_ADDRESS;
DMA_InitStructure.DMA_Memory0BaseAddr    = (uint32_t)&APP_Rx_Buffer;
DMA_InitStructure.DMA_DIR                = DMA_DIR_PeripheralToMemory;
DMA_InitStructure.DMA_BufferSize         = APP_RX_DATA_SIZE;
DMA_InitStructure.DMA_PeripheralInc      = DMA_PeripheralInc_Disable;
```

```
DMA_InitStructure.DMA_MemoryInc          = DMA_MemoryInc_Enable;
DMA_InitStructure.DMA_PeripheralDataSize = DMA_MemoryDataSize_Byte;
DMA_InitStructure.DMA_MemoryDataSize     = DMA_MemoryDataSize_Byte;
DMA_InitStructure.DMA_Mode               = DMA_Mode_Circular;
DMA_InitStructure.DMA_Priority           = DMA_Priority_VeryHigh;
DMA_InitStructure.DMA_FIFOMode           = DMA_FIFOMode_Enable;
DMA_InitStructure.DMA_FIFOThreshold      = DMA_FIFOThreshold_Full;
DMA_InitStructure.DMA_MemoryBurst        = DMA_MemoryBurst_Single;
DMA_InitStructure.DMA_PeripheralBurst    = DMA_PeripheralBurst_Single;
DMA_Init(DMA2_Stream0, &DMA_InitStructure);
DMA_ITConfig(DMA2_Stream0, DMA_IT_TC, ENABLE);
DMA_Cmd(DMA2_Stream0, ENABLE);
Timeout = 100000;
while ((DMA_GetCmdStatus(DMA2_Stream0) != ENABLE) && (Timeout-- > 0));
if (Timeout == 0) {
  while(1);
}
NVIC_InitStructure.NVIC_IRQChannel = DMA2_Stream0_IRQn;
NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;
NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0;
NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
NVIC_Init(&NVIC_InitStructure);
ADC_CommonInitStructure.ADC_Mode = ADC_TripleMode_RegSimult;
ADC_CommonInitStructure.ADC_TwoSamplingDelay =
ADC_TwoSamplingDelay_5Cycles;
```

```
  ADC_CommonInitStructure.ADC_DMAAccessMode =
  ADC_DMAAccessMode_1;
  ADC_CommonInitStructure.ADC_Prescaler = ADC_Prescaler_Div2;
  ADC_CommonInit(&ADC_CommonInitStructure);
  ADC_InitStructure.ADC_Resolution = ADC_Resolution_8b;
  ADC_InitStructure.ADC_ScanConvMode = DISABLE;
  ADC_InitStructure.ADC_ContinuousConvMode = ENABLE;
  ADC_InitStructure.ADC_ExternalTrigConvEdge =
  ADC_ExternalTrigConvEdge_None;
  ADC_InitStructure.ADC_DataAlign = ADC_DataAlign_Right;
  ADC_InitStructure.ADC_NbrOfConversion = 1;
  ADC_Init(ADC1, &ADC_InitStructure);
  ADC_RegularChannelConfig(ADC1, ADC_Channel_12, 1,
  ADC_SampleTime_3Cycles);
  ADC_Init(ADC2, &ADC_InitStructure);
  ADC_RegularChannelConfig(ADC2, ADC_Channel_12, 1,
  ADC_SampleTime_3Cycles);
  ADC_Init(ADC3, &ADC_InitStructure);
  ADC_RegularChannelConfig(ADC3, ADC_Channel_12, 1,
  ADC_SampleTime_3Cycles);
  ADC_MultiModeDMARequestAfterLastTransferCmd(ENABLE);
  is_adc_on = 0;
  for(;;);
}
void cmd_exec_start (void) {
```

```
  if (is_adc_on == 0) {
    ADC_Cmd(ADC1, ENABLE);
    ADC_Cmd(ADC2, ENABLE);
    ADC_Cmd(ADC3, ENABLE);
    ADC_SoftwareStartConv(ADC1);
    APP_Rx_ptr_out = 0;
    APP_Rx_ptr_in = 0;
    is_adc_on = 1;
  }
}
void cmd_exec_stop (void) {
  if (is_adc_on == 1) {
    ADC_Cmd(ADC1, DISABLE);
    ADC_Cmd(ADC2, DISABLE);
    ADC_Cmd(ADC3, DISABLE);
    APP_Rx_ptr_out = 100;
    APP_Rx_ptr_in = 0;
    is_adc_on = 0;
  }
}
```

usbd_cdc.h

```
#ifndef __USBD_CDC_H
#define __USBD_CDC_H
```

```
#include "stm32f4xx.h"
#include "usbd_cdc_core.h"
#include "usbd_conf.h"
#include "stm32f4_discovery.h"
#include <stdio.h>
typedef struct {
  uint32_t bitrate;
  uint8_t  format;
  uint8_t  paritytype;
  uint8_t  datatype;
} LINE_CODING;
void send_data (uint8_t* buf, uint32_t len);
#endif
```

usbd_cdc.c

```
#ifdef USB_OTG_HS_INTERNAL_DMA_ENABLED
#pragma data_alignment = 4
#endif
#include "usbd_cdc.h"
#include "usb_cdc_cmd.h"
uint8_t cdc_ByteRx = 0x00;
LINE_CODING linecoding = {
  10500000, /* baud rate 10500000 */
  0x00,     /* stop bits-1*/
```

```
  0x00,     /* parity - none*/
  0x08      /* nb. of bits 8*/
};
extern uint8_t  APP_Rx_Buffer [];
extern uint32_t APP_Rx_ptr_in;
extern uint32_t APP_Rx_ptr_out;
static uint16_t cdc_Init   (void);
static uint16_t cdc_DeInit (void);
static uint16_t cdc_Ctrl   (uint32_t Cmd, uint8_t* Buf, uint32_t Len);
static uint16_t cdc_DataTx (uint8_t* Buf, uint32_t Len);
static uint16_t cdc_DataRx (uint8_t* Buf, uint32_t Len);
CDC_IF_Prop_TypeDef cdc_fops = {
  cdc_Init,
  cdc_DeInit,
  cdc_Ctrl,
  cdc_DataTx,
  cdc_DataRx
};
static uint16_t cdc_Init(void) {
  STM32F4_Discovery_LEDInit(LED6);
  STM32F4_Discovery_LEDOn(LED6);
  APP_Rx_ptr_in = 0;
  APP_Rx_ptr_out = 0;
  return USBD_OK;
}
```

```
static uint16_t cdc_DeInit(void) {
return USBD_OK;
}
static uint16_t cdc_Ctrl (uint32_t Cmd, uint8_t* Buf, uint32_t Len) {
switch (Cmd) {
case SEND_ENCAPSULATED_COMMAND:
  break;
case GET_ENCAPSULATED_RESPONSE:
  break;
case SET_COMM_FEATURE:
  break;
case GET_COMM_FEATURE:
  break;
case CLEAR_COMM_FEATURE:
  break;
case SET_LINE_CODING:
  linecoding.bitrate    = (uint32_t)(Buf[0] | (Buf[1] << 8) | (Buf[2] << 16) | (Buf[3]
  << 24));
  linecoding.format     = Buf[4];
  linecoding.paritytype = Buf[5];
  linecoding.datatype   = Buf[6];
  break;
case GET_LINE_CODING:
  Buf[0] = (uint8_t)(linecoding.bitrate);
  Buf[1] = (uint8_t)(linecoding.bitrate >> 8);
```

}

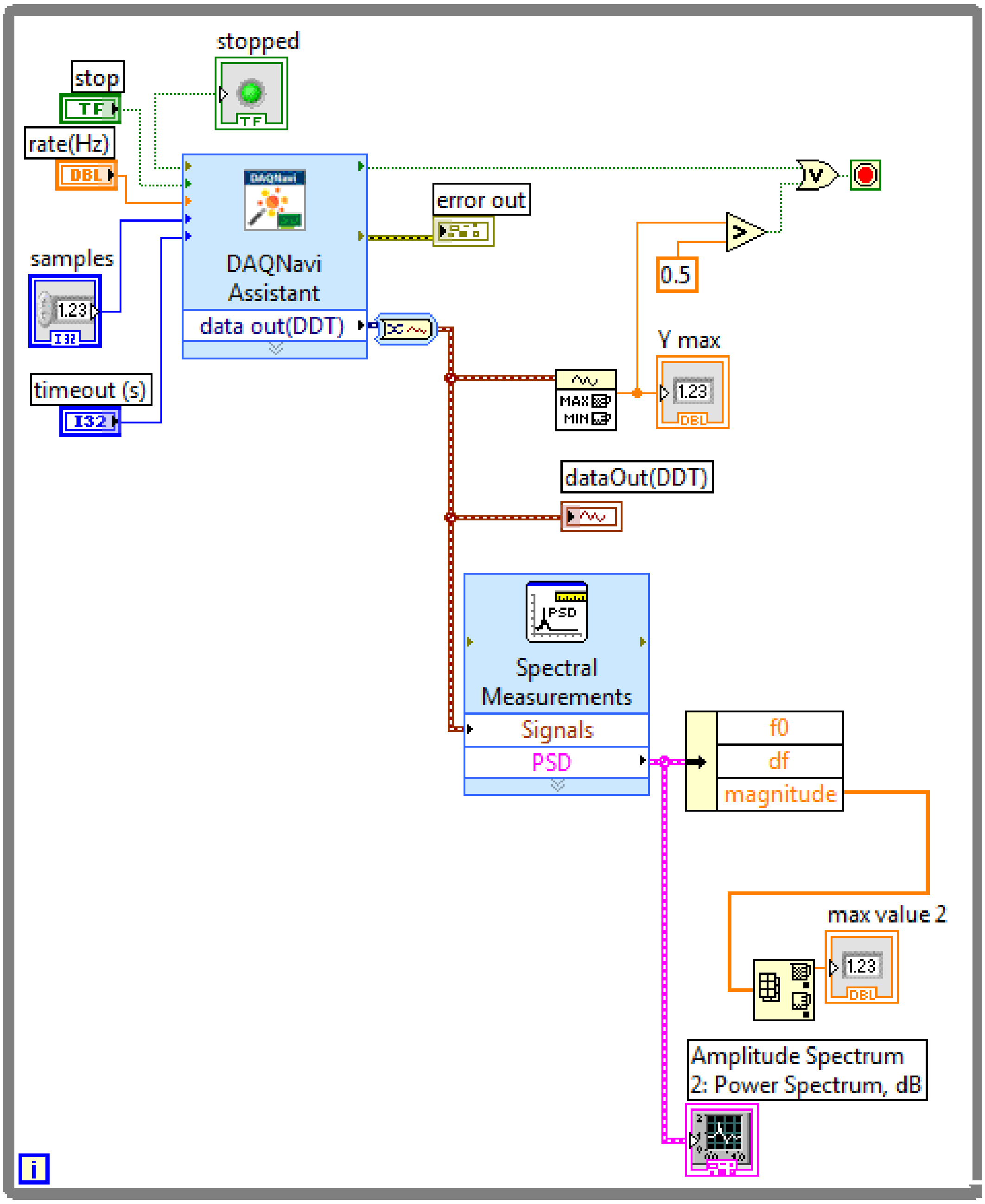

```
    Buf[2] = (uint8_t)(linecoding.bitrate >> 16);
    Buf[3] = (uint8_t)(linecoding.bitrate >> 24);
    Buf[4] = linecoding.format;
    Buf[5] = linecoding.paritytype;
    Buf[6] = linecoding.datatype;
    break;
  case SET_CONTROL_LINE_STATE:
    break;
  case SEND_BREAK:
    break;
  default:
    break;
  }
  return USBD_OK;
}
static uint16_t cdc_DataTx (uint8_t* Buf, uint32_t Len) {
  uint32_t i;
  for(i = 0; i < Len; i++) {
    APP_Rx_Buffer[APP_Rx_ptr_in++] = Buf[i];
    if (APP_Rx_ptr_in == APP_RX_DATA_SIZE) {
      APP_Rx_ptr_in = 0;
    }
  }
  return USBD_OK;
}
```

```
static uint16_t cdc_DataRx (uint8_t* Buf, uint32_t Len) {
  uint32_t i;
  for (i = 0; i < Len; i++) {
    cdc_ByteRx = *(Buf + i);
    if (cdc_ByteRx == '@') {
      cmd_exec_start();
      cdc_ByteRx = '\0';
    }
  }
  return USBD_OK;
}
void send_data (uint8_t* buf, uint32_t len) {
  cdc_DataTx (buf, len);
}
```

main.c

```
#include "stm32f4xx.h"
#include "stm32f4_discovery.h"
#include "usb_cdc_cmd.h"
int main (void) {
  SystemInit();
  cmd_exec_function();
}
```

Appendix C - LabVIEW DAQ and GUI for Verification



Appendix D – AE Sensor Specification

R80α Sensor

General Purpose, 800 kHz Resonant Frequency Acoustic Emission Sensor

Description and Features

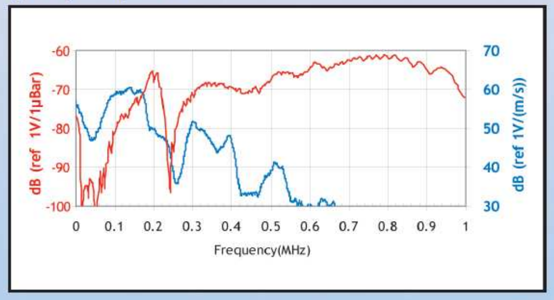
The Alpha series family of sensors features SMA connectors versus the Microdot connectors found on PAC's RXX series of passive sensors. The Alpha series includes R3α, R6α, R15α, R30α R50α, R80α and WSα sensors. The major improvements in Alpha series over the RXX series include:

- Use of the more popular SMA type of connector.
- Cavity is machined from a solid stainless steel rod making for a simpler and more robust design.
- Dramatically increased thickness of the ceramic shoe for better mechanical stability.
- Distance from the bottom of the ceramic shoe to the bottom edge of sensor cavity increased for better insulation resistance and ground avoidance.
- Introduced a 30-degree angle at the bottom edge of the sensor cavity.

All these improvements make the Alpha series sensors more robust, reliable and greatly reduce the possible grounding of the cavity to the structure caused by wet environment.

Application

High frequency AE sensors such as the R80α are often used in a high noise environments on applications such as brittle crack detection and processing of AE signals with high frequency components.



Frequency response of the R80α. Calibration based on ASTM E1106; Calibration based on ASTM E976.

Operating Specifications

Dynamic

Peak Sensitivity V/(m/s); [V/µbar] ..... 58 [-62] dB

Operating Frequency Range ....200 - 1000 kHz

Resonant Freq. V/(m/s); [V/µbar] .... 200 [800] kHz

Directionality.................................... ±1.5 dB

Environmental

Temperature Range ....-65 to 175°C

Shock Limit .... 500 g

Completely enclosed crystal for RFI/EMI immunity

Physical

Dimensions...... 0.75" dia. x 0.84" h (19 x 21.4 mm)

Weight..32 grams

Case Material............................ Stainless Steel

Face Material...................................Ceramic

Connector... SMA

Connector Locations ....Side

Seal .... Epoxy

Sensor to Preamp Cable (1 or 2 meters)..... 1232-X-SMA

Ordering Information and Accessories

R80α .... R80α or R80a

Magnetic Hold-Down MHR15A

Preamplifier......................................0/2/4, 2/4/6

Preamp to System Cable (specify length in meters)..... 1234 - X

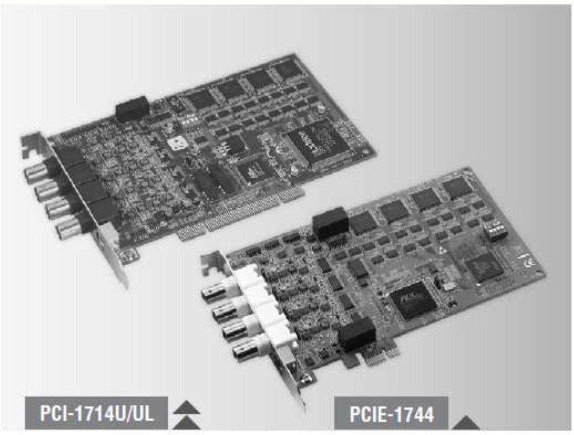
Sensors include

NIST Calibration Certificate & Warranty

Appendix E – DAQ Board Specification

PCI-1714U
PCI-1714UL
PCIE-1744

30 MS/s, 12-bit, Simultaneous 4-ch Analog Input Universal PCI Card
10 MS/s, 12-bit, Simultaneous 4-ch Analog Input Universal PCI Card
30 MS/s, 12-bit, Simultaneous 4-ch Analog Input PCI Express Card



Features

- 4 single-ended analog input channels
- 12-bit A/D converter, with up to 30 MHz sampling rate
- Programmable gain
- Onboard FIFO memory (PCI-1714U/PCIE-1744: 32,768 samples; PCI-1714UL: 8,192 samples, each channel)
- 4 A/D converters simultaneously sampling
- Multiple A/D triggering modes
- Programmable pacer/counter
- BoardID™ switch
- Universal PCI Bus (PCI-1714U/PCI-1714UL) (supports 3.3 V or 5 V PCI bus signals)
- PCI Express V1.0 (PCIE-1744)

Introduction

PCI-1714U/PCI-1714UL and PCIE-1744 are advanced high-performance data acquisition cards based on the PCI/PCIe bus. With a large FIFO of 32,768 for each channel, the maximum sampling rate of PCI-1714U/PCIE-1744 can get up to 30 MS/s, on each channel, with an emphasis on continuous, non-stop, high-speed, streaming data of samples to host memory. The low-cost PCI-1714UL offers 10 MS/s on each channel at a stable rate, and has also been equipped with a universal PCI interface.

Specifications

Analog Input

- **Channels**: 4 single-ended
- **Resolution**: 12 bits
- **Max. Sampling Rate**: PCI-1714U/PCIE-1744: 30 MS/s; PCI-1714UL: 10 MS/s
- **FIFO Size**: PCI-1714U/PCIE-1744: 32,768 samples each channel; PCI-1714UL: 8,192 samples each channel
- **Overvoltage Protection**: PCI-1714U/PCI-1714UL: 30 Vp-p; PCIE-1744: 14 Vp-p
- **Input Impedance**: 50 Ω/1 MΩ/Hi Z jumper selectable/100 pF
- **Sampling Modes**: Software polling, pacer
- **Trigger Modes**: Post-trigger, pre-trigger, delay-trigger, about-trigger
- **Input Range (V)**: ±5, ±2.5, ±1, ±0.5

General

- **Bus Type**: PCI-1714U/PCI-1714UL: Universal PCI V2.2; PCIE-1744: PCI Express V1.0
- **I/O Connectors**: 4 x BNC connector (for AI); 1 x PS/2 connector (for Ext. clock and trigger)
- **Dimensions (L x H)**: 175 x 100 mm (6.9" x 3.9")
- **Power Consumption**: Typical: 5 V @ 850 mA ; 12 V @ 600 mA; Max.: 5 V @ 1 A; 12 V @ 700m A
- **Operating Temperature**: 0 ~ 60° C (32 ~ 140° F)
- **Storage Temperature**: -20 ~ 85° C (-4 ~ 185° F)
- **Storage Humidity**: 5 ~ 95% RH, non-condensing (refer to IEC 68-2-3)

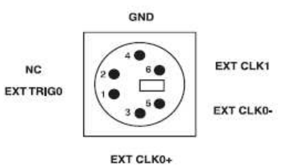
Ordering Information

- **PCI-1714U**: 30 MS/s, 12-bit, Simultaneous 4-ch AI PCI Card
- **PCI-1714UL**: 10 MS/s, 12-bit, Simultaneous 4-ch AI PCI Card
- **PCIE-1744**: 30 MS/s, 12-bit, Simultaneous 4-ch AI PCIe Card

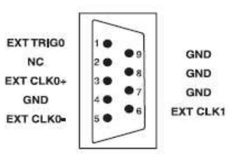
Accessories

- **ADAM-3909**: DB9 DIN-rail Wiring Board
- **PCL-1010B-1**: BNC to BNC Wiring Cable, 1 m
- **PCL-10901-1**: DB9 to PS/2 Cable, 1 m
- **PCL-10901-3**: DB9 to PS/2 Cable, 3 m

Pin Assignments



Onboard PS/2 Connector



PS/2 To DB9 Cable Connector

Appendix F – Xbee Specification

Key Features

High Performance, Low Cost

XBee

- Indoor/Urban: up to 133' (40 m)
- Outdoor line-of-sight: up to 400' (120 m)
- Transmit Power: 2 mW (3 dBm)
- Receiver Sensitivity: -96 dBm

XBee-PRO (S2)

- Indoor/Urban: up to 300' (90 m), 200' (60 m) for International variant
- Outdoor line-of-sight: up to 2 miles (3200 m), 5000' (1500 m) for International variant
- Transmit Power: 50mW (17dBm), 10mW (10dBm) for International variant
- Receiver Sensitivity: -102 dBm

XBee-PRO (S2B)

- Indoor/Urban: up to 300' (90 m), 200' (60 m) for International variant
- Outdoor line-of-sight: up to 2 miles (3200 m), 5000' (1500 m) for International variant
- Transmit Power: 63mW (18dBm), 10mW (10dBm) for International variant
- Receiver Sensitivity: -102 dBm

Advanced Networking & Security

Retries and Acknowledgements

DSSS (Direct Sequence Spread Spectrum)

Each direct sequence channel has over 65,000 unique network addresses available

Point-to-point, point-to-multipoint and peer-to-peer topologies supported

Self-routing, self-healing and fault-tolerant mesh networking

Low Power

XBee

- TX Peak Current: 40 mA (@3.3 V)
- RX Current: 40 mA (@3.3 V)
- Power-down Current: < 1 µA

XBee-PRO (S2)

- TX Peak Current: 295mA (170mA for international variant)
- RX Current: 45 mA (@3.3 V)
- Power-down Current: 3.5 µA typical @ 25 degrees C

XBee-PRO (S2B)

- TX Peak Current: 205mA (117mA for international variant)
- RX Current: 47 mA (@3.3 V)
- Power-down Current: 3.5 µA typical @ 25 degrees C

Easy-to-Use

No configuration necessary for out-of box RF communications

AT and API Command Modes for configuring module parameters

Small form factor

Extensive command set

Free X-CTU Software (Testing and configuration software)

Free & Unlimited Technical Support

Worldwide Acceptance

FCC Approval (USA) Refer to Appendix A for FCC Requirements. Systems that contain XBee®/XBee-PRO® ZB RF Modules inherit Digi Certifications.

ISM (Industrial, Scientific & Medical) 2.4 GHz frequency band

Manufactured under ISO 9001:2000 registered standards

XBee®/XBee-PRO® ZB RF Modules are optimized for use in US, Canada, Europe, Australia, and Japan (contact Digi for complete list of agency approvals).

Specifications

Specifications of the XBee®/XBee-PRO® ZB RF Module

Specification	XBee	XBee-PRO (S2)	XBee-PRO (S2B)
Performance			
Indoor/Urban Range	up to 133 ft. (40 m)	Up to 300 ft. (90 m), up to 200 ft (60 m) international variant	Up to 300 ft. (90 m), up to 200 ft (60 m) international variant
Outdoor RF line-of-sight Range	up to 400 ft. (120 m)	Up to 2 miles (3200 m), up to 5000 ft (1500 m) international variant	Up to 2 miles (3200 m), up to 5000 ft (1500 m) international variant
Transmit Power Output	2mW (+3dBm), boost mode enabled 1.25mW (+1dBm), boost mode disabled	50mW (+17 dBm) 10mW (+10 dBm) for International variant	63mW (+18 dBm) 10mW (+10 dBm) for International variant
RF Data Rate	250,000 bps	250,000 bps	250,000 bps
Data Throughput	up to 35000 bps (see chapter 4)	up to 35000 bps (see chapter 4)	up to 35000 bps (see chapter 4)
Serial Interface Data Rate (software selectable)	1200 bps - 1 Mbps (non-standard baud rates also supported)	1200 bps - 1 Mbps (non-standard baud rates also supported)	1200 bps - 1 Mbps (non-standard baud rates also supported)
Receiver Sensitivity	-96 dBm, boost mode enabled -95 dBm, boost mode disabled	-102 dBm	-102 dBm
Power Requirements			
Supply Voltage	2.1 - 3.6 V	3.0 - 3.4 V	2.7 - 3.6 V
Operating Current (Transmit, max output power)	40mA (@ 3.3 V, boost mode enabled) 35mA (@ 3.3 V, boost mode disabled)	295mA (@3.3 V) 170mA (@3.3 V) international variant	205mA, up to 220 mA with programmable variant (@3.3 V) 117mA, up to 132 mA with programmable variant (@3.3 V), International variant
Operating Current (Receive))	40mA (@ 3.3 V, boost mode enabled) 38mA (@ 3.3 V, boost mode disabled)	45 mA (@3.3 V)	47 mA, up to 62 mA with programmable variant (@3.3 V)
Idle Current (Receiver off)	15mA	15mA	15mA
Power-down Current	< 1 uA @ 25°C	3.5 µA typical @ 25°C	3.5 µA typical @ 25°C
General			
Operating Frequency Band	ISM 2.4 GHz	ISM 2.4 GHz	ISM 2.4 GHz
Dimensions	0.960" x 1.087" (2.438cm x 2.761cm)	0.960 x 1.297 (2.438cm x 3.294cm)	0.960 x 1.297 (2.438cm x 3.294cm)
Operating Temperature	-40 to 85° C (industrial)	-40 to 85° C (industrial)	-40 to 85° C (industrial)
Antenna Options	Integrated Whip Antenna, Embedded PCB Antenna, RPSMA, or U.FL Connector	Integrated Whip Antenna, Embedded PCB Antenna, RPSMA or U.FL Connector	Integrated Whip Antenna, Embedded PCB Antenna, RPSMA or U.FL Connector
Networking & Security			
Supported Network Topologies	Point-to-point, Point-to-multipoint, Peer-to-peer, and Mesh	Point-to-point, Point-to-multipoint, Peer-to-peer, and Mesh	Point-to-point, Point-to-multipoint, Peer-to-peer, and Mesh
Number of Channels	16 Direct Sequence Channels	14 Direct Sequence Channels	15 Direct Sequence Channels
Channels	11 to 26	11 to 24	11 to 25
Addressing Options	PAN ID and Addresses, Cluster IDs and Endpoints (optional)	PAN ID and Addresses, Cluster IDs and Endpoints (optional)	PAN ID and Addresses, Cluster IDs and Endpoints (optional)
Agency Approvals			
United States (FCC Part 15.247)	FCC ID: OUR-XBEE2	FCC ID: MCQ-XBEEPRO2	FCC ID: MCQ-PROS2B
Industry Canada (IC)	IC: 4214A-XBEE2	IC: 1846A-XBEEPRO2	IC: 1846A-PROS2B
Europe (CE)	ETSI	ETSI (International variant)	ETSI (10 mW max)

Specifications of the XBee®/XBee-PRO® ZB RF Module

Specification	XBee	XBee-PRO (S2)	XBee-PRO (S2B)
Australia	C-Tick	C-Tick	C-Tick
Japan	R201WW07215215 Wire, chip, RPSMA, and U.FL versions are certified for Japan. The PCB antenna version is not.	R201WW08215142 (international variant) Wire, chip, RPSMA, and U.FL versions are certified for Japan. PCB antenna version is not.	R201WW10215062 (international variant)
RoHS	Compliant	Compliant	Compliant

Hardware Specs for Programmable Variant

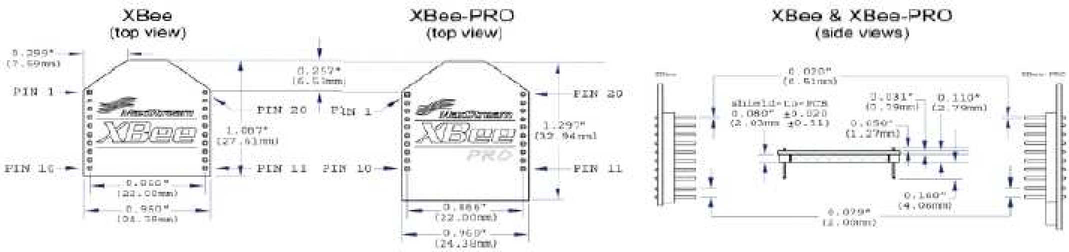
The following specifications need to be added to the current measurement of the previous table if the module has the programmable secondary processor. For example, if the secondary processor is running and constantly collecting DIO samples at a rate while having the RF portion of the XBEE sleeping the new current will be $I_{total} = I_{r2} + I_0$, where I_{r2} is the runtime current of the secondary processor and I_s is the sleep current of the RF portion of the module of the XBEE-PRO (S2B) listed in the table below.

Specifications of the programmable secondary processor

Optional Secondary Processor Specification	These numbers add to S2B specifications (Add to RX, TX, and sleep currents depending on mode of operation)
Runtime current for 32k running at 20MHz	+14mA
Runtime current for 32k running at 1MHz	+1mA
Sleep current	+0.5uA typical
For additional specifications see Freescale Datasheet and Manual	MC9S08QE32
Minimum Reset low pulse time for EM250	+50 nS (additional resistor increases minimum time)
VREF Range	1.8VDC to VCC

Mechanical Drawings

Mechanical drawings of the XBee®/XBee-PRO® ZB RF Modules (antenna options not shown)



Pin Signals

Pin Assignments for the XBee/XBee-PRO Modules
(Low-asserted signals are distinguished with a horizontal line above signal name.)

Pin #	Name	Direction	Default State	Description
1	VCC	-	-	Power supply
2	DOUT	Output	Output	UART Data Out
3	DIN / $\overline{\text{CONFIG}}$	Input	Input	UART Data In
4	DIO12	Both	Disabled	Digital I/O 12
5	$\overline{\text{RESET}}$	Both	Open-Collector with pull-up	Module Reset (reset pulse must be at least 200 ns)
6	RSSI PWM / DIO10	Both	Output	RX Signal Strength Indicator / Digital IO
7	DIO11	Both	Input	Digital I/O 11
8	[reserved]	-	Disabled	Do not connect
9	$\overline{\text{DTR}}$ / SLEEP_RQ/ DIO8	Both	Input	Pin Sleep Control Line or Digital IO 8
10	GND	-	-	Ground
11	DIO4	Both	Disabled	Digital I/O 4
12	$\overline{\text{CTS}}$ / DIO7	Both	Output	Clear-to-Send Flow Control or Digital I/O 7. CTS, if enabled, is an output.
13	ON / $\overline{\text{SLEEP}}$	Output	Output	Module Status Indicator or Digital I/O 9
14	VREF	Input	-	Not used for EM250. Used for programmable secondary processor. For compatibility with other XBEE modules, we recommend connecting this pin voltage reference if Analog sampling is desired. Otherwise, connect to GND.
15	Associate / DIO5	Both	Output	Associated Indicator, Digital I/O 5
16	$\overline{\text{RTS}}$ / DIO6	Both	Input	Request-to-Send Flow Control, Digital I/O 6. RTS, if enabled, is an input.
17	AD3 / DIO3	Both	Disabled	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Both	Disabled	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Both	Disabled	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0 / Commissioning Button	Both	Disabled	Analog Input 0, Digital IO 0, or Commissioning Button

- Signal Direction is specified with respect to the module
- See Design Notes section below for details on pin connections.

Appendix G – Paper Publication

ICBIR 2014

International Conference on Business and Industrial Research

Certificate of Participation

We hereby certify that the paper entitled:

The Implementation of Cost-Effective Data Acquisition System for Acoustic Emission Sensor using Variable Gain Preamplifier and STM32F4 Microcontroller Interface

authored by:

Chavadol Mekchar, Wipawadee Wongsuwan, Warakorn Srichavengsup, Wimol San-Um

have participated in the International Conference on Business and Industrial Research (ICBIR 2014),

held at Thai-Nichi Institute of Technology, Bangkok, Thailand, May 15-16, 2014

Associate Professor Dr. Pichit Sukchareonpong

Editor-in-Chief

The Implementation of Cost-Effective Data Acquisition System for Acoustic Emission Sensor using Variable Gain Preamplifier and STM32F4 Microcontroller Interface

Chavadol Mekchar*, Wipawadee Wongsuwan**, Warakorn Srichavengsup*, Wimol San-Um*

*Intelligent Electronic System Research Laboratory
**Energy Management Laboratory, Research Center of Advanced Energy Technology
Faculty of Engineering, Thai-Nichi Institute of Technology
Pattanakarn 37, Saunlaung, Bangkok, Thailand, 10250
Tel: (+662) 763-2600 Ext. 2926, Fax: 0-2763-2700, E-mail: me.chavadol_st@tni.ac.th; wimol@tni.ac.th.

***Abstract*— this paper presents the Data Acquisition (DAQ) system that particularly collects Acoustic Emission (AE) signals in the range of 1 kHz to 500 kHz from the sensor, and transmits to a personal computer for further signal analyses. The proposed system comprises two major components, i.e. a variable gain preamplifier and a STM32F4 microcontroller. The variable gain preamplifier is implemented using BiMOS operational amplifier model CA3140 which provides relatively high input impedance and high-speed performance. The signal is subsequently digitized by STM32F4 Microcontroller and transmits to data analysis software via serial communication (RS232). Simulations are performed through Orcad PSpice and a MATLAB Simulink. Signal quality analysis involves frequency response, Total Harmonic Distortion (THD), and Power Spectral Density (PSD). Experiments show that the proposed system offers a potential alternative to commercially available data acquisition systems.**

Keywords— Data Acquisition System, Acoustic Emission Sensor, Variable Gain Preamplifier, STM32F4 Microcontroller.

I. INTRODUCTION

Machine condition monitoring and fault diagnosis system are necessary tools for the condition base maintenance. The input signals for these systems are generated by various kinds of sensors and transducers such as temperature, pressure, flow, vibration, and especially Acoustic Emission (AE) signal [1]. Acoustic emission is a sound wave, which is generated when a material undergoes stress due to an external force. The AE signal is a phenomenon occurring in, for example, mechanical loading generating sources of elastic waves. Such occurrence is the result of a slight surface displacement of materials produced from stress waves, which is generated when the energy in materials or on its surface is released rapidly [2].

In order to obtain AE signal for purpose of monitoring, the Data Acquisition (DAQ), this is a process of measuring an electrical or physical phenomenon such as voltage or current. Fig.1 illustrates the typical acoustic emission sensor detection

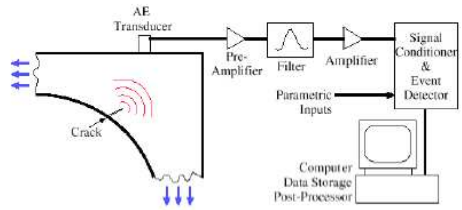


Fig. 1. Typical acoustic emission sensor detetcion system.

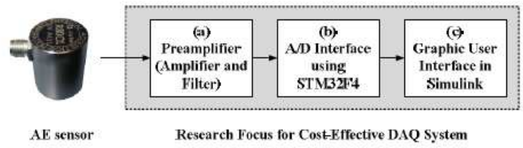
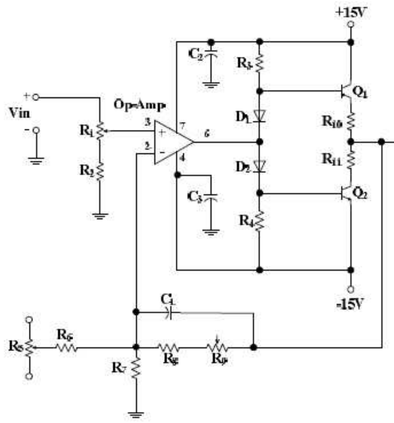
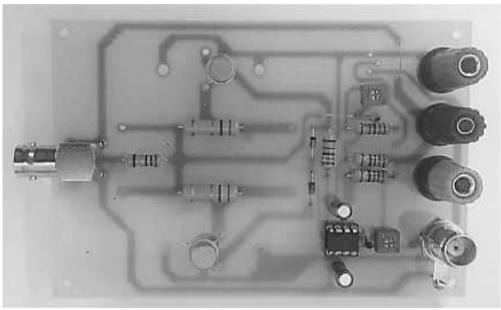


Fig. 2. Overall System Designs with research focus on cost-effective DAQ Board

system where AE transducer provides AE signals to the preamplifier, filter, amplifier, signal conditioning and event detector [3,4]. Such a typical system is relatively complicated as numerous devices are required. Typically, the DAQ system consists of sensor input, measurement hardware, and a computer with programmable software. Despite the fact that commercially available DAQ system exploits the processing power, productivity, display, and connectivity capabilities of industry-standard computers with high effectiveness, the price is relatively costly and may not be suitable for some applications especially Small and Medium Enterprises (SME). Considerations on the typical acoustic emission sensor detection system shown Fig.1 and general requirements for DAQ system used in AE sensing system lead to the possibility to simplify the system using simple, but robust devices.



(a)



(b)

Fig. 3. The preamplfier circuit designs; (a) circuit diagram, (b) circuit implemenation on board.

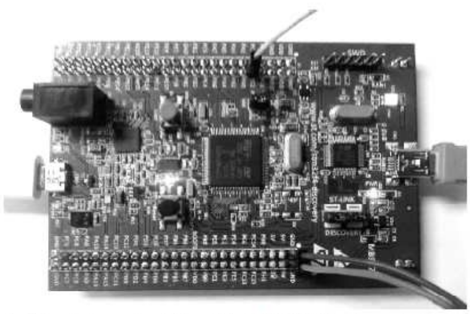


Fig. 4. The photograph of the utilised STM32F4 microcontroller as an interface device for A/D converter.

Although there has been numerous research reports for AE signal feature characterizations that include commercially available DAQ system [5-10], no particular research focuses has been investigated on a simple-but-robust DAQ system. This paper therefore proposes a new cost-effective DAQ

TABLE I.
SUMMARY OF PREAMPLIFIER CIRCUIT PARAMETERS AND PART NUMBERS

Preamplifier Circuit Parameters					
Resistors		Capacitors		Integrated Circuits	
Parts	Values	Parts	Values	Parts	Numbers
R_1	2kΩ	C_1	2pF	Op-Amp	CA3140
R_2	200Ω	C_2	50μF	Diode D_1	1N914
R_3	2.2kΩ	C_3	50μF	Diode D_2	1N914
R_4	2.2kΩ	-	-	Transistor Q_1	2N3053
R_5	10kΩ	-	-	Transistor Q_2	2N4037
R_6	3kΩ	-	-	-	-
R_7	200Ω	-	-	-	-
R_8	2.2kΩ	-	-	-	-
R_9	50kΩ	-	-	-	-
R_{10}	2.7Ω	-	-	-	-
R_{11}	2.7Ω	-	-	-	-
R_{12}	51Ω	-	-	-	-

system in which a preamplifier and a filter is implemented through BiMOS operational amplifier model CA3140 providing relatively high input impedance and high-speed performance. Analog-to-Digital (A/D) converter is implemented by Real-time Operating System (ROS) hardware. Signal quality analysis will include time-domain analysis, frequency response, Total Harmonic Distortion (THD), and Power Spectral Density (PSD). Simulations will be performed using Orcad PSpice and a MATLAB Simulink. Experiments will achieved by an integrated circuit on a prototype board and STM32F4 microcontroller

II. PROPOSED COST-EFFECTIVE DATA ACQUISITION SYSTEM FOR AE SIGNAL DETETCION

A. *Overall System Designs*

Fig. 2 summarizes overall System Designs with research focus on cost-effective DAQ Board. The proposed system consists of three major blocks. First, an integrated amplifier and filter, which is so called preamplifier, that is used to amplify the input signal with variable gain and filter noises in environment. Second, the A/D converter interface using STM32F4 controller which operates realtime and has an ability to connect to computer through MATLAB program. Last, the graphic user interface system exploits MATLAB display capability to show the realtime-signal and various signal characteristics analysis. In this paper, the AE signal is considered to be a sinusoidal signal, which represents the AE signal acoustic at a single harmonic signal, for analysing the DAQ characteristics.

B. *Pre-Amplifier Circuit Descriptions*

Figs.3 (a) and (b) show the preamplifier circuit designs. The variable gain preamplifier is implemented using BiMOS operational amplifier model CA3140 as a high slew rate, wideband amplifier. The amplifier circuit is designed base on a non-inverting amplifier which analog input signal is connected at the input terminal which signal level can be adjusted by R_1 before input of CA3140. Gain of preamplifier can be adjusted via R_9 and output DC level can also be adjusted by R_5 to keep output signal at the output terminal in

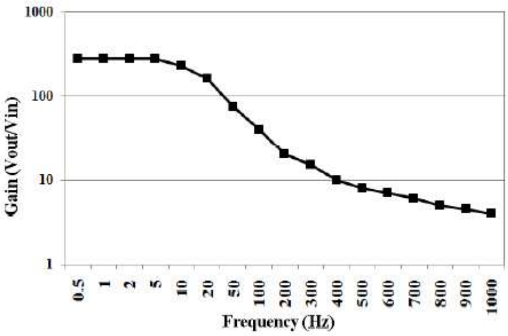


Fig. 5. Frequency characteristics of the proposed preamplifier

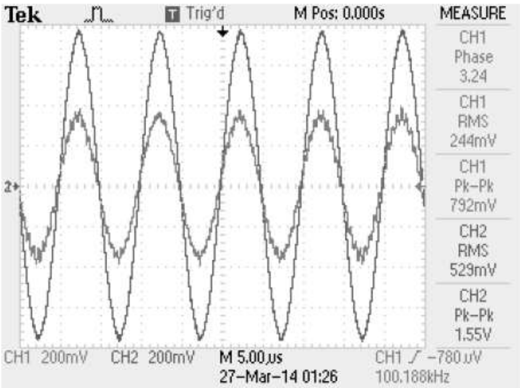


Fig. 6. Illustration of sinusoidal singnal in time domain.

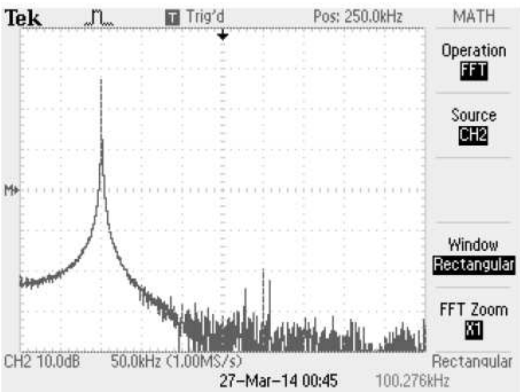


Fig. 7. Illustration of the Fast Fourier Transform for THD calculations.

the range of 0-5 V that compatible with STM32F4 Controller. The push-pull class AB output circuit using transistors Q_1 and Q_2 are exploited in order to adjust full peak-to-peak swing.

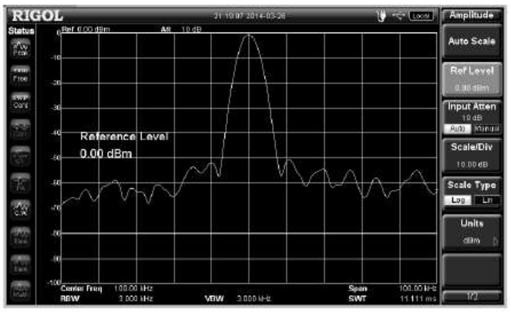


Fig. 8. Illustrations of the power spectral density showing the bandwith of the signal at 0dBm level.

C. *STM32F4 Microcontroller Interface Operations*

As mentioned earlier, this paper utilizes the STM32F4 controller, which is based on the high-performance ARM®Cortex™-M4 32-bit RISC core operating at a frequency of up to 168 MHz, for analog-to-digital conversion. This device offers three 12-bit ADCs, two DACs, a low-power RTC, twelve general-purpose 16-bit timers including two PWM timers for motor control, two general-purpose 32-bit timers. Therefore, the input analog signal is digitized with the performance of 3×12-BIT AMD 2.4 MSPS.

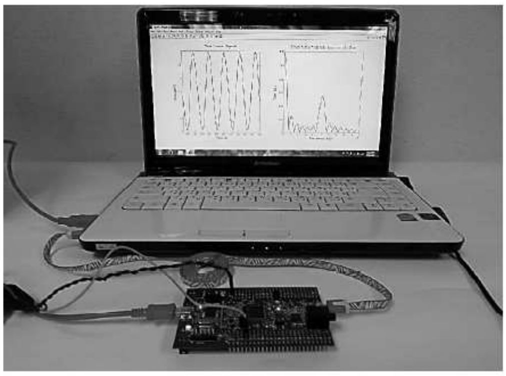
III. Experimental Results

The experimental results have been performed through the protoboard corresponding to the block diagram in Fig. 2. The demonstration condition is a sinusoidal signal input of 20mV at 100 kHz, corresponding to the low-frequency signals. Fig. 5 shows frequency characteristics of the proposed preamplifier. It can be seen from the graph that the preamplifier establish the low pass filter characteristic where the low frequency has a very high gain of above 100 but at the high frequency, for example 1 MHz, the gain is still greater than 0. For the frequency of interest at 100 kHz the gain is approximately at 20dB. Fig. 6 illustrates the sinusoidal signal in time domain. It can be seen that the non-inverting amplifier can amplify the input signal to the larger signal without phase shift.

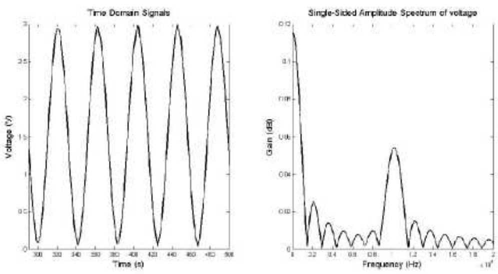
Fig. 7 illustrates of the Fast Fourier Transform for THD calculations. Using the rectangular window transformation, the peak signal at 100 kHz has the highest peak where the noise floor is about 200 kHz far from the center frequency. Since FFT transformation was used for the THD calculation, Table 2 summarized the THD in percentage at difference value of particular gain under region of interest between 5-20. It is seen that the THD is in the range of 3-5%. This is acceptable for the preamplifier that amplifies the AE signal in real application. Fig. 8 illustrates the power spectral density showing the bandwidth of the signal at 0dBm level. It is seen that the noise floor is approximately -60dB and the bandwidth is about 4 kHz. Fig. 9 demonstrates the STM32F4 and the graphic user interface; (a) Experimental setup, (b) the display of signal in time and frequency domains.

TABLE II.
SUMMARY OF PARTICULAR GAINS (V_{OUT}/V_{IN}) AND SHOWING THE CORRESPONDING THD IN PERCENTAGES

Particular Gains	THD (%)
5	3.2362
10	3.7334
15	5.4146
20	5.6488



(a)



(b)

Fig. 9. Demonstration of STM32F4 and the graphic user interface; (a) Experiemntal setup, (b) the display of signal in time and frequency domains.

IV. CONCLUSIONS

This paper has presented the data acquisition system that particularly collects AE signals in the range of 1 kHz to 500 kHz from the sensor, and transmits to a personal computer for further signal analyses. The proposed system comprises two major components, i.e. a variable gain preamplifier and a STM32F4 Microcontroller. The variable gain preamplifier is implemented using BiMOS operational amplifier model CA3140 which provides relatively high input impedance and high-speed performance. It is obviously seen through the results that the quality signal analysis were strongly satisfy including high frequency response at high voltage gain, obtain the peak of signal at 100 kHz. The generated signal was successfully digitized by STM32F4 Microcontroller and transmits to a MATLAB Simulink via serial communication (RS232). In addition, the proposed system offers a potential alternative cost-effective data acquisition to commercially available data acquisition systems in the same range of frequency response such as National Instrument (NI), Physical Acoustics Corporation (PAC) etc. in which the cost could be reduced to approximately 80% comparing to the specified DAQs.

ACKNOWLEDGEMENTS

The authors are grateful to Research and Academic Services Division of Thai-Nichi Institute of Technology for financial supports. The authors would like to thanks Mr.Bandhit Suksiri and Mr.Natthorn Chuayphan for their helps in microcontroller implementations.

REFERENCES

[1] Krzysztof Jemielniak, "Some aspects of acoustic emission signal pre-processing" Journal of Materials Processing Technology 109 (2001) 242-247

[2] Jiao Yang, Yang Qingxin, Li Guanghai, Zhang Jingyan, "Acoustic emission source identification technique for buried gas pipeline leak" IEEE 2006 1-4244-0342-1/06

[3] M. Noipitak, A. Prateepasen, W. Kaewwaewnoi, "A Relative Calibration Method for Valve Leakage Rate Measurement System" Elsevier Measurement 44 (2011) 211-218

[4] Didem Ozevin, James Harding, "Novel Leak Localization in Pressurized Pipeline Networks using Acoustic Emission and Geometric Connectivity" International Journal of Pressure Vessels and Piping 92 (2012) 63-69

[5] Athanasios Anastasopoulos, Dimitrios Kourousis, Konstantinos Bolas, "Acoustic emission leak detection of liquid filled buried pipeline" Acoustic Emission Group (2009) J. Acoustic Emission, 27

[6] Valeriy N. Ovcharuk, Qin Hongwu, "Hardware-Software Complex for Acoustic Emission Spectral Analyzing" International Siberian Conference on Control and Communications SIBCON (2011), 218-220

[7] A. Prateepasen, W. Kaewwaewnoi, P. Kaewtrakulpong, "Smart Portable Noninvasive Instrument for Detection of Internal Air Leakage of a Valve using Acoustic Emission Signals" Elsevier Measurement 44 (2011) 378-384

[8] N.C. Hii, C.K. Tan, S.J. Wilcox, Z.S. Chong, "An Investigation of the Generation of Acoustic Emission from the Flow of Particulate Solids in Pipelines" Elsevier Powder Technology 243 (2013) 120-129

[9] A. Mostafapour, S. Davoudi, "Analysis of Leakage in High Pressure Pipe using Acoustic Emission Method" Elsevier Applied Acoustics 74 (2013) 335-342

[10] R. Hou, A. Hunt, R.A. Williams, "Acoustic monitoring of pipeline flows: particulate slurries" Elsevier Powder Technology 106 (1999) 30–36