Chapter 1

Introduction

1.1 Background of the Study

Enhancing the capacity and quality of multiuser wireless communications system through the use of antenna arrays has been given enough attention in space-time signal processing techniques [1]. Array signal processing is a part of signal processing that uses sensors that are organized in patterns, or arrays, to detect signals and to determine information about them and the most common application of this is in the detection of acoustic signals with microphones serving as the sensors [2].

In any microphone array processing it always involves two main procedures, sound source localization and beamforming. Source localization is the task of locating a sound source given measurements of the sound field. Sound field can either be a sound pressure or a particle velocity. Beamforming, on the other hand, focuses the microphone array "beam" towards the source.

Many studies have already been conducted for source localization using the direction-of-arrival (DOA) estimation [3-22]. Knowing the exact location of a target using DOA estimation received much interest for the past decade because of the wide range of applications it offers such as in radar, sonar, communication, speech processing, oceanography and navigation [3]. There are several algorithms being implemented in order to compute for the spectral estimation of an incoming signal over a noisy channel. Among of those are the correlation, multiple signal classification (MUSIC), maximum likelihood estimator (MLE), estimation of signal parameters using rotational invariance techniques (ESPRIT) and matrix pencil [23]. Since it is the most promising and a leading candidate for further study and for the actual hardware implementation, MUSIC algorithm will be given more attention in this study. MUSIC algorithm is one of the high-resolution subspace-based methods which compute the DOA by dividing the cross-correlation matrix of the array signals into signal and noise subspaces through eigenvalue decomposition.

With its wide range of applications, actual implementation in hardware of DOA is undeniably given much focus right now. But due to its computational complexity in real-time processing and high cost of implementation, it encounters some restrictions for its successful implementation. Digital signal processing (DSP) plays an important role in this study. Almost all of the real-world applications today involve DSP ranging from high-definition TV, mobile telephony, digital audio, multimedia, digital cameras, radar, sonar detectors, biomedical imaging, global positioning, digital radio and speech recognition [24].

Much research has been conducted to develop DSP algorithms and systems for real-world applications. More recently, the field-programmable gate array (FPGA) has been proposed as a hardware technology for DSP systems as they offer the capability to develop the most suitable circuit architecture for the computational, memory and power requirements of the application. Presently, FPGAs have been gaining considerable attention in high-performance DSP applications, and are emerging as coprocessors for standard DSP processors that need specific accelerators. FPGAs offer a tremendous computational power by using highly parallel architectures for very high performance [25].

1.2 Statement of the Problem

In a typical wireless environment wherein many signal sources could be expected, determining the exact location of each signal is significantly considered. In some cases wherein closely spaced signal sources are present, knowing the number of possible sources is also vital. This is most prominent in radar applications wherein the position of the moving target is being determined and most radar systems use parabolic reflector antennas for signal transmission and detection.

Because of the increasing demand for more accurate and better performance in target detection, a much newer technology called the phase array technique is being introduced. With this technique, it uses an antenna composed of many elements such as dipoles which could be implemented as either a planar array or a line array. Each element of a phased array is phase shifted to steer the beam to the desired direction [26]. This technique can provide a robust and better detection performance than the reflectors but the cost of actual implementation of this is expensive.

As for the Philippines, there are no such studies devoted primarily for signal detection using antenna array. There are a lot of studies already done abroad regarding DOA estimation in antenna array but most of these investigations only involve software simulations to show the result. Only few have implemented it in actual hardware for real-time processing because of the complexity being involved. Since DOA estimation offers very promising applications, an actual hardware implementation using FPGA will be carried out in this study.

1.3 Significance of the Study

Using microphone array in determining for the DOA of sound source has some benefits. In a video conferencing or in a long distance TV-based classroom, DOA estimates can be used to automatically steer cameras to the speaker. On the other hand, having the sound signal captured from several points, it allows for spatial filtering (also called beamforming) if being given proper processing which could result to the amplification of signal originating from the specific directions, or the beam, and attenuating signals from other directions. This seems to be promising for the hearing aids of those people with deficiency in hearing as they could perceive better the origin of a sound in their surroundings. Speech enhancement for human computer interfaces that depend on speech inputs from operators, human machine interaction for robots and acoustic surveillance systems are also among the benefits of knowing the DOA.

This study aims to implement a uniform linear array using microphone sensors which will be tested in a small, closed room. Knowing the fact that in a closed room, reverberation of sound signal could be most expected which could result to high selective fading. Using an antenna array in this kind of environment performs better than using a single antenna only. Several sound sources positioned in various locations will be set and the estimation of the direction of their arrival will be made. To compute for the DOA, MUSIC algorithm will be implemented using an FPGA. Once successfully carried out, this would also be a good alternative for radar system using parabolic reflectors used in military applications wherein accurate target detection is vital.

1.4 Objectives of the Study

The main objective of this study is to implement an algorithm that will estimate the direction-of-arrival of a sound source using a linear array of microphone sensors. Software simulations will be performed to validate the effectiveness of the algorithm and to test for its veracity in real-life applications, hardware implementation will be done using FPGA.

To achieve the main goal of this research, the following specific objectives will be considered:

- Construct a four-element uniform linear antenna array using microphones as its sensors to locate the position of a sound source.
- Develop an external board that will store the sampled data from each sensor and will synchronize the data transfer from each sensor to the FPGA board for parallel signal processing.
- Process the three main steps of MUSIC algorithm (i.e. correlation matrix estimation, eigenvalue decomposition and spectral estimation) in the FPGA board to meet the real-time processing demands.
- Use Verilog HDL as the hardware description language for the algorithm to be utilized in the FPGA.
- Evaluate the system's performance and draw some recommendations for reference in some future studies.

1.5 Scope and Delimitations

The following are the scope and delimitations of this study:

- For the simplicity of actual implementation, only four (4) sensors will be used and placed linearly in equal distance. Although, the more number of sensors to be used the more accurate the data will be but the cost and complexity of implementing it is in trade-off.
- Since four sensors will be used, only four analog-to-digital converters (ADC) will also be used with each sensor having its own ADC.
- Since wideband MUSIC is considered in this study, it should follow the rule that the number of signal sources is less than the number of sensors to avoid any ambiguity in its performance [3]. Therefore, only three (3) sound sources will be taken into account for this particular study.
- Uniform linear array (ULA) has no capability of knowing the signals from its front or back side, or signals directly above or below it. Thus, no signal source will be placed on the said positions.
- Only acoustic signal will be given interest in this study for the simplicity of hardware implementation. However, sound signals are also comparable to radio frequency (RF) signals. The difference is that the latter operate at a much higher frequency which means a more complex circuit is needed for its realization.
- Actual testing of the prototype will be done inside a closed room only. Outdoor testing is not recommended for it introduces more noise that could severely affect the system.
- Altera Cyclone IV E F484 FPGA board will be utilized for the signal processing.
- Verilog HDL will be employed as the hardware description language to program the MUSIC algorithm in the FPGA board.

1.6 Definition of Terms

Algorithm : an effective method expressed as a finite list of well-defined instructions for calculating a function

Beamforming : a general signal processing technique used to control the directionality of the reception or transmission of the signal

Correlation matrix : it computes the correlation coefficients of the columns of the matrix; that is row i and column j of the correlation matrix is the correlation between the row i and column j of the original matrix and its diagonal elements will be one since they are the correlation of a column with itself

Eigen decomposition : (or sometimes called spectral decomposition) is the factorization of a matrix into a canonical form, whereby the matrix is represented in terms of its eigenvalues and eigenvectors

Field-programmable gate array : is an integrated circuit designed to be configured by the customer or designer after manufacturing.

Hardware description language : any language from a class of computer languages, specification languages, or modelling languages for formal description and design of electronic circuits and most commonly digital logic.

Reverberation : the persistence of sound in a particular space after the original sound is removed; it is created when a sound is produced in an enclosed space causing a large number of echoes to build up and then slowly decay as the sound is absorbed by the walls and air

Selective fading : fading resulting from multipath propagation varies with frequency since each frequency arrives at the receiver in different time

Sound pressure : (acoustic pressure) the local pressure deviation from the ambient atmospheric pressure caused by a sound wave

Spectral estimation : estimation of the spectral density (also known as the power spectrum) of a random signal from a sequence of time samples of the signal

Wideband : a relative term used to describe a wide range of frequencies in a spectrum. A system is typically wideband if the message bandwidth exceeds the channel's coherence bandwidth.

Chapter 2

Review of Related Literature and Studies

This part discusses some related studies and fundamental concepts which are helpful to fully understand the whole research. Several ideas by different authors have been gathered to strongly support the framework of the study and to prove its feasibility. The review is done per topic area.

2.1 Antenna Array Implementation

Array signal processing is a part of signal processing that uses sensors in patterns or arrays to detect signals and determine information about the signal [27]. Antenna arrays can be arranged in either linear or circular (see fig. 2.1) and mostly are uniform wherein the spacing of each sensor is the same.

Because the ULA is one dimensional, there is a surface of ambiguity on which it is unable to determine information about signals. It suffers from 'front-back ambiguity,' meaning that signals incident from 'mirror locations' at equal angles on the front and back sides of the array are undistinguishable (see fig. 2.2) [2].

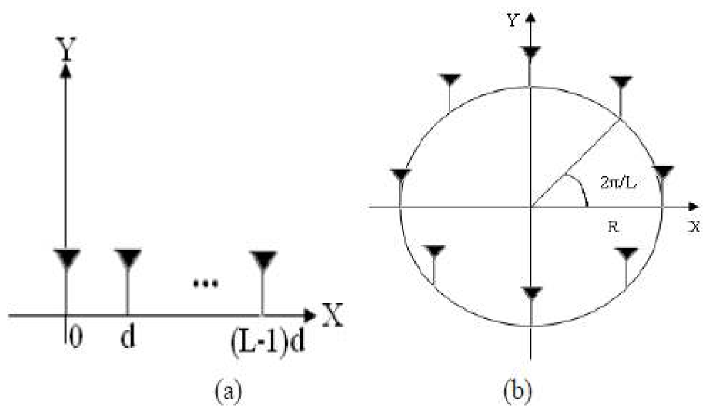

Figure 2.1. Antenna array arrangement. (a) uniform linear array and (b) uniform circular array [27]

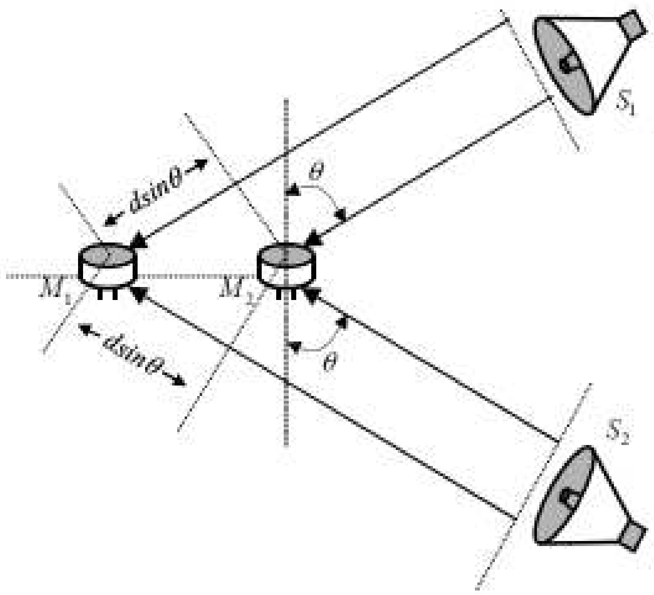

Figure 2.2. Uniform linear array with front-back ambiguity [28].

To know the basic principles lying in a uniform linear antenna array, consider the figure shown in figure 2.3 below. The array is made up of four microphones placed in a straight line with equal distance, *d*, between adjacent microphones.

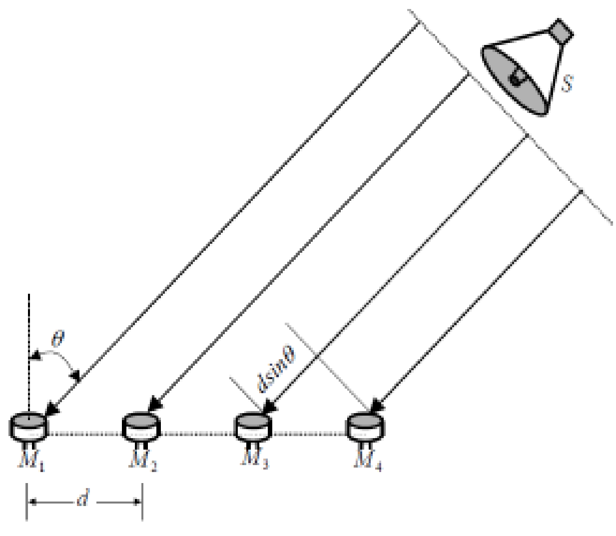

Figure 2.3. Uniform linear array with far field source [28].

In this case, *S* or the distance of the source is assumed to be greater than the distance between each sensor which means that it is in the far field array. With this assumption, we can approximate the spherical wavefront that originates from the source as a plane wavefront. Therefore, it can be said that the sound waves arriving at each microphones are parallel to each other. The direction perpendicular to the array is called the broadside direction or sometimes called the broadside of the array which will be the reference of the measurement of DOA.

As with light waves, acoustic waves exhibit the same physical relationship. The frequencies of signals that an array detects are important because they determine constraints on the spacing of the sensors. The array's sensors sample incident signals in space and, just as aliasing occurs in analog to digital conversion when the sampling rate does not meet the Nyquist criterion, aliasing can also happen in space if the sensors are not sufficiently close together [2].

A useful property of the ULA is that the delay from one sensor to the next is uniform across the array because of their equidistant spacing. Trigonometry reveals that the additional distance the incident signal travels between sensors is $d\sin(\theta)$. Thus, the time delay between consecutive sensors is given by:

$$\tau = \frac{d}{c}\sin(\theta) \tag{2-1}$$

If f_{max} will be the highest frequency to be used, phase differences between spatially sampled signals should be set to π or less because phase differences above π cause incorrect time delays to be seen between received signals to avoid spatial aliasing. Therefore, a condition below should be satisfied,

$$2\pi\tau f_{max} \leq \pi \tag{2-2}$$

Substituting for τ in (2-1) yields

$$d \leq \frac{c}{2f_{\max}\sin(\theta)} \tag{2-3}$$

The worst delay occurs for $\theta = 90^{\circ}$, so we can get

$$d \leq \frac{\lambda_{\min}}{2} \tag{2-4}$$

For the distance between sensors to avoid signals aliasing in space,

$$\lambda_{\min} = \frac{c}{f_{\max}} \tag{2-5}$$

Many studies about DOA estimation have been conducted by using equally-spaced linear array [3-4, 10, 17, 20] and equally-spaced circular array [7, 19].

2.2 Electret Microphone

Electret microphones are the most commonly used microphones today. Every cellphone and laptop has one embedded into it, and many studio microphones are also electrets. They can have an extremely wide frequency response (from 10Hz to 30 kHz), and typically cost less than a dollar. They are also very small and quite sensitive [29]. An electret microphone is a type of condenser microphone which eliminates the need for a polarizing power supply by using a permanently charged material.

Figure 2.4. A common electret microphone [29].

It is well known that condenser microphones are the transducer of choice when accuracy, stability, frequency characteristics, dynamic range, and phase are important. However, conventional condenser microphones require critical and expensive implementation as well as the need for the high DC bias for linearity. Because of these disadvantages, it paved a way for the development of electret microphones in early 1960s at Bell Labs for the improved linearity in communications.

KS Lee and SP Lee [14] used eight electrets condenser microphones to find the listener's position using the sound signal from the listener. It was implemented together with a microprocessor board, a DSP board and a multi-channel ADC board and an analog frontend.

2.3 Direction of Arrival Estimation

The direction-of-arrival (DOA) estimation problem has been extensively studied in the signal processing literature. In many practical signal processing problems, the objective is to estimate from measurements a set of constant parameters upon which the received signals depend. High-resolution DOA estimation is important in many sensor systems such as radar, sonar electronic surveillance and seismic exploration. More recently, it shows another importance in some useful applications such as design and control of robots and large flexible space structures [30].

Estimating the direction of multiple signals in noise is an important problem in signal processing. Several approaches can be used for this problem which include subspace techniques such as Multiple Signal Classification (MUSIC), adaptive techniques such as Least Mean-Square (LMS) estimation, or fast implementations of discrete Fourier transform (DFT) such as the Goertzel algorithm. The choice of the technique is based on a trade-off between the observation time and the available computational resources [31].

Out of the wide range of algorithms for DOA estimation, MUSIC has been widely studied. In a detailed evaluation based on thousands of simulations, M.I.T.'s Lincoln Laboratory concluded that, among currently accepted high-resolution algorithms, MUSIC was the most promising and a leading candidate for further study and actual hardware implementation. However, although the performance advantages of MUSIC are substantial, they are achieved at a considerable cost in computation (searching over parameter space) and storage (of array calibration data) [30].

There are several variants of MUSIC being used in order to improve the performance of the estimation. Root-MUSIC, cyclic-MUSIC and smooth-MUSIC are among the types. In a study conducted by Gao, Nallanathan and Wang [12], they have implemented a modified form of MUSIC algorithm to improve the DOA estimation accuracy as well as increase the maximum number of detectable signals. They have successfully proved the effectiveness of their proposed algorithm by considering both circular and noncircular signal sources.

The ESPRIT algorithm also receives much attention in the area of DOA estimation. Shahbazpanahi, Valaee and Bastani [9] utilized such algorithm in the estimation of the central angle and angular extension of distributed sources. They have found out that their proposed algorithm can be applied to a multisource scenario in which different sources may have different parametric angular power densities. On the other hand, this algorithm could also be implemented underwater as the study of Tichavsky, Wong and Zoltowski [32] showed that their new underwater acoustic eigenstructure ESPRIT-based algorithm using a single vector hydrophone outperforms an array of spatially displaced pressure hydrophones. However, as compared with MUSIC, ESPRIT presents significantly greater computation load.

Aside from MUSIC and ESPRIT, maximum likelihood (ML) is also making its way in spotlight. Bergamo et al. [8] utilized the approximate maximum likelihood (AML) for source localization and DOA estimation. Besides computing for the DOA, ML is also employed in estimating for the Doppler shift produced by a nonstationary source on an antenna array.

2.4 The MUSIC Algorithm for DOA estimation

MUSIC algorithm is a high resolution technique for DOA estimation based on the exploiting of the eigenstructure of the covariance matrix. Among the information that it can provide are the number of the incident signals, the DOA of each signal, noise power, etc [48].

If we have M array elements spaced by half wavelength, the signal model received can be written as a linear combination of the D incident waveforms from far field and white Gaussian noise. The matrix representation is

$$\mathbf{X}(n) = \mathbf{A} \cdot \mathbf{S}(n) + \mathbf{N}(n) \tag{2-6}$$

Where *S(n)* is a signal vector, *X(n)* is a *M x 1* vector of array output at any sampling time n, which is also called snapshot and *N(n)* is the noise vector. The columns of A = [$\alpha(\theta_1)$, $\alpha(\theta_2)$.... $\alpha(\theta_D)$] are the steering vectors.

Considering MUSIC for the detection of the DOA of a source signal, the following will be the steps to be used [38]:

1. Estimate the correlation matrix with received vector X(n) and correlation can be written by

$$\mathbf{R}_\mathrm{X} = E\left\{X(n)X^H(n)\right\} = A \cdot \mathbf{R}_\mathrm{S} \cdot A^H + \sigma^2\mathbf{I} \tag{2-7}$$

   where *E[•]* and the superscript ^{H} denote expectation and complex conjugate transpose, respectively. R_S = $E[S(n)S^H(n)]$ is the signal correlation matrix and σ^2 is the noise variance.

2. Perform eigenvalue decomposition of R_X to obtain eigenvalues $\lambda_1, \lambda_{2\ldots,} \lambda_M$ and its corresponding eigenvectors $e_1, e_{2\ldots,} e_M$.
3. DOA estimation can be performed using either of the following two representations:

   Noise subspace method,

$$P(\theta) = \frac{1}{a^H(\theta)GG^H a(\theta)} \tag{2-8}$$

   Signal subspace method,

$$P(\theta) = \frac{1}{a^H(\theta)\left(I - SS^H\right)a(\theta)} \tag{2-9}$$

4. The D signal directions are the D largest peaks of $P(\theta)$.

One important thing to consider about MUSIC algorithm is that, it works only very well at high signal-to-noise ratio (SNR). When SNR decreases, its performance is greatly affected. Also, since MUSIC is a subspace method, it should be observed that the number of sources is less than the number of sensors.

2.5 Beamforming

Beamforming is the process of combining sounds or electromagnetic signals that comes from only one particular direction and impinges different sensors at the receiver. As being a measure of directivity, this important concept is used in different communication, voice and sonar applications [27]. A beamformer combines the spatially distributed sensor collected array data linearly with the beamforming weight to achieve spatial filtering. Beamforming enhances the signal from the desired spatial direction and reduces the signal(s) from other direction(s) in addition to possible time/frequency filtering. The beamformer output is a coherently enhanced

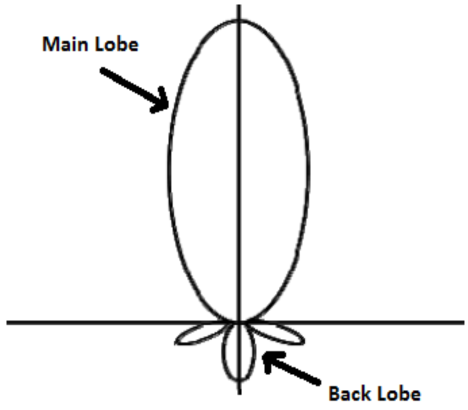

Figure 2.5. Visualization of a practical beamformer [27]

estimate of the transmitted signal, with one set of weights for each source. In many cases, the desired signal direction may need to be estimated [15].

If ULA is used, the output of each sensor will be the same except that each one will be delayed by different amount. All the outputs will be added together and the signal propagating through the array will reinforce while cancelling the noise. The canonical form of the time invariant beamformer in frequency domain is just a weighted sum:

$$Y(f) = \sum_{m=1}^{M-1} W_m(f) X_m(f) \tag{2-10}$$

where $X_m(f)$ is the signal, captured from i-th microphone, $Y(f)$ is the beamformer output and $W_m(f)$ are the time invariant frequency dependent weights. With properly designed weights we can aim the beam to given direction, reducing the ambient noise and reverberation. While fast and reliable, the time invariant beamformer has limited performance (noise suppression and directivity index) [33]. More sophisticated adaptive beamforming algorithms are known in the research community [4, 5, 8, 19, 20].

2.6 Sound Source Localization

There are several approaches to determine the direction to the sound source.

Time delay estimates (TDE)-based methods use the fact that the sound reaches the microphones with slightly different times. The delays are easily computed using cross-correlation function between the signals from different microphones. Variations of this approach use different weighting (maximum likelihood, PHAT, etc.) to improve the reliability and stability of the results under noise and reverberation conditions [33].

Today, most microphone arrays use more than two microphones to find the overall direction. Finding the direction from all possible pairs and averaging it does not work well in case of reverberation. The most common method is testing the hypothesis for direction of arrival using the sum of all cross-correlation functions with proper delays.

Another approach is to steer the beam and to compute the direction based on the maximum output signal. This method gives similar results to time delay estimates with ML weighting. In all cases post-processing of the sound source localization results is critical. Various methods are used, ranging from particle filtering to real-time clustering. The goal of the post-processor is to remove accidental reflections and reverberations, leaving the results from one or more sound sources [33].

2.7 Real-Time Digital Signal Processing

There are two types of digital signal processing (DSP) applications: non-real time and real-time. According to the book of Kuo, Lee and Tian [25], non-real time signal processing involves manipulating signals that have already been collected in digital forms. This may or may not represent a current action, and the requirement for the processing result is not a function of real-time. Real-time signal processing places stringent demands on DSP hardware and software designs to complete predefined tasks within a certain time frame.

2.7.1 Analog Interface

The process of converting an analog signal to a digital signal is called the analog-to-digital conversion, usually performed by an analog-to-digital converter (ADC). The purpose of signal conversion is to prepare real-world analog signals for processing by digital hardware. Once the input digital signal has been processed by the DSP hardware, the result $y(n)$ is still in digital form. In many DSP applications, we need to reconstruct the analog signal after the completion of digital processing. We must convert the digital signal $y(n)$ back to the analog signal $y(t)$ before it is applied to an appropriate analog device. This process is called the digital-to-analog conversion, typically performed by a digital-to-analog converter (DAC).

2.7.2 The Basics of Sampling

The ADC converts the analog signal $x(t)$ into the digital signal $x(n)$. Analog-to-digital conversion, commonly referred as digitization, consists of the sampling (digitization in time) and quantization (digitization in amplitude) processes as shown in figure 2.6. The sampling process depicts an analog signal as a sequence of values. The basic sampling function can be carried out

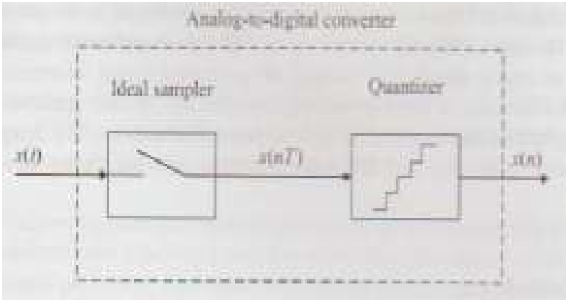

Figure 2.6. Block diagram of an ADC [25].

with an ideal sample-and-hold circuit, which maintains the sampled signal level until the next sample is taken. Next, the quantization process approximates a waveform by assigning a number for each sample. Therefore, the analog-to-digital conversion performs the following steps:

1. The band limited signal $x(t)$ is sampled at uniformly spaced instants of time nT, where n is a positive integer and T is the sampling period in seconds. This sampling converts an analog signal into a discrete–time signal $x(nT)$ with continuous amplitude value.
2. The amplitude of each discrete-time sample is quantized into one of the 2^B levels, where B is the number of bits that the ADC has to represent for each sample. The discrete amplitude levels are represented (or encoded) into distinct binary words $x(n)$ with a fixed wordlength B.

The reason for making this distinction is that the processes introduce different distortions. The sampling process brings in aliasing or folding distortion, while the encoding process results in quantization noise. The sampling period is defined as

$$T = \frac{1}{fs} \tag{2-11}$$

where f_s is the sampling frequency (or sampling rate) in hertz (or cycles per second). The intermediate signal *x(nT)* is a discrete-time signal with a continuous value at discrete time *nT, n* = 0, 1...∞ (see figure 2.7). In order to represent an analog signal *x(t)* by a discrete-time *x(nT)* accurately, the sampling frequency must be at least twice the maximum frequency component (f_M) in analog signal *x(t)*. That is

$$fs \geq 2f_M \tag{2-12}$$

Where f_M is also called the bandwidth of the signal *x(t)*. The minimum sampling rate $f_s = 2f_M$ is called the Nyquist rate. The frequency $f_N = f_s/2$ is called the Nyquist frequency or the folding frequency.

2.7.3 DSP Hardware

DSP systems are required to perform intensive arithmetic operations such as multiplication and addition. These tasks maybe implemented on microprocessors, microcontrollers, digital signal processors, or custom integrated circuits. Although it is possible to implement DSP algorithms on any digital computer, the real applications determine the optimum hardware platform. According to Kuo, Lee and Tian [25], there are five hardware platforms widely used for DSP systems.

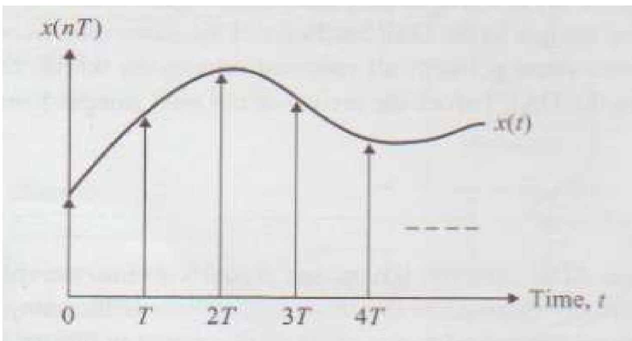

Figure 2.7. Example of analog signal *x(t)* and discrete-time signal *x(nT)* [25].

1. Special-purpose (custom) chips such as application-specific integrated circuits (ASIC)
2. Field-programmable gate arrays (FPGA)
3. General-purpose microprocessors or microcontrollers
4. General-purpose digital signal processors (DSP processors) and
5. DSP processors with application-specific hardware (HW) accelerators

ASIC devices are usually designed for specific tasks that require a lot of computations such as digital subscriber loop (DSL) modems, or high-volume products that use mature algorithms such as Fast Fourier transform and Reed-Solomon codes. FPGAs have been used in DSP applications for years as glue logics, bus bridges and peripherals for reducing system costs and affording a higher level of system integration. Recently, FPGAs have been gaining much attention in high-performance DSP applications, and are emerging as coprocessors for standard DSP processors that need specific accelerators. In the study of ZhangXin [34], a custom-designed FPGA is used for complex weighting in a four-element DBF multibeam antenna system and showed remarkable results.

General-purpose μP/μC becomes faster and increasingly able to handle some DSP applications. General architectures of μP/μC are the Harvard architecture and von Neumann architecture. On the other hand, a DSP processor is basically a microprocessor optimized for processing repetitive numerically intensive operations at high rates. Today, DSP processors have become the foundation of many new markets beyond the traditional signal processing areas for technologies and innovations in motor and motion control, automotive systems, home appliances, consumer electronics, and vast range of communication systems and devices [25]. In a study conducted by K. S. Lee and S. P. Lee [14], they have both utilized microprocessor and

DSP in the cancellation of the cross-talk signals at an arbitrary listener position in a stereophonic playback system.

2.8 FPGA-Based Hardware Implementation

Digital signal processing (DSP) is used in a wide range of applications. Recently, the field-programmable gate array (FPGA) has been proposed as a hardware technology for DSP systems as they offer the capability to develop the most suitable circuit architecture for the computational, memory and power requirements of the application in a similar way to system-on-chip (SoC) systems [24].

According to Woods, McAllister, Yi and Lightbody [24], FPGAs emerged as simple 'glue logic' technology, providing programmable connectivity between major components where the programmability was based on either antifuse, EPROM or SRAM technologies. By simply reprogramming the FPGA, design errors can be corrected thus allowing the interconnectivity of the components to be changed as required. Because of this ability, FPGA provides a new level of reassurance in an increasingly competitive market and has been widely used today.

Some studies have implemented variants of MUSIC algorithm in FPGA [35-38]. Zou, Hongyuan and Guowen [35] have implemented an improved MUSIC algorithm in FPGA with high-speed parallel optimization. The new algorithm offered a lower computation cost compared to the original MUSIC but at the expense of slight decrease in performance. On the other hand, unitary MUSIC algorithm was being implemented in an FPGA-based DOA estimator in the studies carried out by Kim, Ichige and Arai [36, 37]. The algorithm was used for cellular wireless base station. Meanwhile, FPGA-based multiple sub-array beam-space MUSIC was the focus in the study of Yao, Li, Zhou, Chen and Yu [38]. Song and Zhang [39], on the other hand, have also implemented MUSIC algorithm in FPGA using eight-element antenna array.

Many other researches have dealt with the FPGA implementation in DOA estimation [40-42]. It includes the development of an FPGA-based high-speed fast fourier transform (FFT) processor [43] and a fuzzy logic system in improving the DOA estimation [44].

Due to the increasing popularity of FPGA, there are several companies who sell such technology which include Xilinx, Altera, Atmel, Lattice and Atmel [24]. In this study, MUSIC algorithm will be fully implemented in FPGA board using Altera Cyclone IV EP4CE55F23C8N (see figure 2.8). It consists of 55, 856 logic elements, 2,340 kilobits of embedded memory, 154 embedded multipliers and 4 phase-locked loops (PLLs). On board, it can support up to 100 user I/Os and 374 I/Os for the device. This device can support the fast computational requirement for the real-time processing of the sound signal.

Figure 2.8. An Altera cyclone FPGA board.

2.9 Verilog HDL as the Programming Language

Programming languages such as FORTRAN, Pascal and C were being used for a very long time for computer programs but they are sequential in nature. In terms of digital design field, there is also a need for designers to have standard languages that will describe digital circuits that is why hardware description languages (HDLs) were developed. HDLs allowed designers to model the concurrency of processes found in hardware elements [45]. Verilog HDL and VHDL (VHSIC Hardware Description Language) are two HDL which became popular. HDLs include ways of describing the propagation of time and signal dependencies (sensitivity) which set them different from programming languages.

Before, HDL-based designs need to be manually translated first into a schematic circuit with interconnections between gates. Because of the advent of logic synthesis, digital circuits could now be described at a register transfer level (RTL) by use of HDL. With this, the designer had only to specify how data flows between registers and how the design processes the data and the details of gates and their interconnections to implement the circuit were automatically extracted.

HDLs have shown also its great impact in the simulation of systems boards, interconnect busses, FPGAs (Field Programmable Gate Arrays) and PALs (Programmable Array Logic) [45].

Verilog HDL has evolved as a standard hardware description language originated in 1983 at the Gateway Design Automation and was invented by Phil Moorby and Prabhu Goel. According to Palnitkar [45], Verilog HDL offers many useful features for hardware design. Among of which are the following:

a. Verilog HDL is a general-purpose hardware description language that is easy to learn and easy to use. It has a similar syntax to the C programming language.
b. Verilog HDL allows different levels of abstraction to be mixed in the same model. Thus, a designer can define a hardware model in terms of switches, gates, RTL or behavioural code.
c. Most popular logic synthesis tools support Verilog HDL.
d. All fabrication vendors provide Verilog HDL libraries for postlogic synthesis simulation. Thus, designing a chip in Verilog HDL allows the widest choice of vendors.
e. The Programming Language Interface (LPI) is a powerful feature that allows the user to write custom C code to interact with the internal data structures of Verilog.

A Verilog design consists of a hierarchy of modules. A module is the basic building block in Verilog. A module can be an element or a collection of lower-level design blocks. A module provides the necessary functionality to the higher-level block through its port interface (inputs and outputs), but hides the internal implementation. This allows the designer to modify module internals without affecting the rest of the design [45].

In a study done by Li [50], he used Verilog HDL to simulate his proposed new advanced encryption standard (AES) to perform both encryption and decryption with 128-, 192-, and 256-bit key options by a novel on-the-fly key generation module. The architecture was implemented in FPGA and ASIC designs. Meanwhile, Verilog was used to implement the very large scale integration (VLSI) intellectual property of genetic algorithm (GA) in FPGA in the study of Chen et al. [51]. Their proposed algorithm dynamically performed various functions to meet the real-time requirements of various GA applications. There are still a lot of studies which have used Verilog HDL as the language of choice for their hardware implementation.

Chapter 3

Methodology

3.1 Conceptual Framework

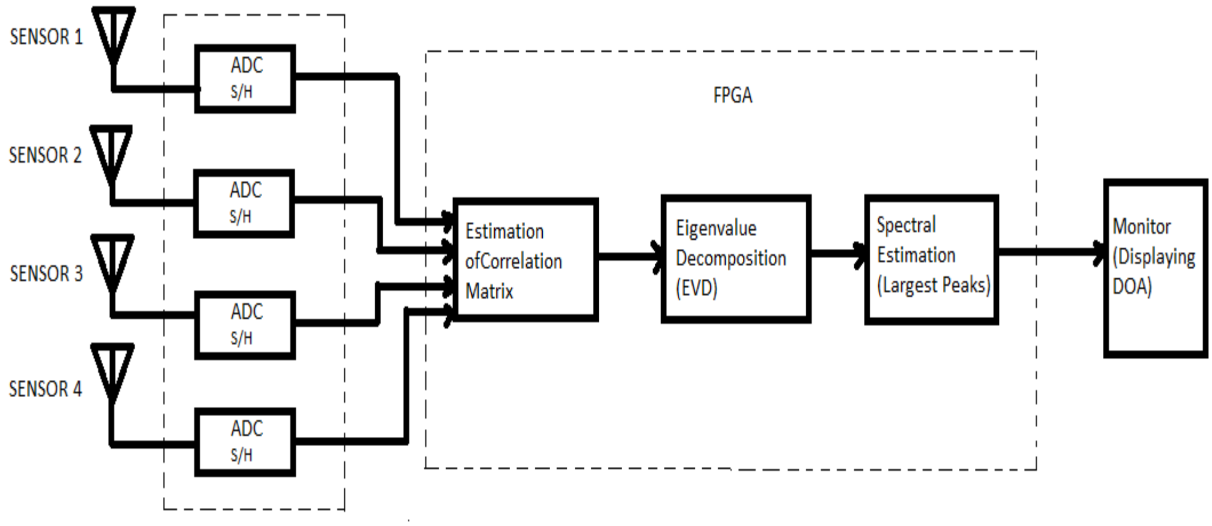

Figure 3.1. Simplified block diagram of the system.

Figure above shows the conceptual framework of the study. Four sensors will be used for the antenna array. These sensors are the electret microphones spaced equally in a linear array. A separate board consisting of four ADC circuit will be utilized. The signals coming from the four sensors will be processed first by the ADC. Each signal will be sampled first and will be stored in the sample-and-hold circuit before the next sample comes in. The sampled signal will be quantized and converted to digital signals to be processed by the FPGA. Data transfer to the FPGA from the sensors will be done simultaneously.

MUSIC algorithm will be employed in this study in order to compute for the signal source's direction. It involves three major stages. First, all the four signals will be combined and the correlation matrix will be approximated. Then, the correlation matrix will be decomposed into its eigenvectors and eigenvalues. To determine the specific position of a certain source, largest peaks developed based from the spectral estimation of the signal will be identified as the

DOA. An LCD monitor or a laptop will be connected at the output port of the FPGA to display the estimated direction.

3.2 Matlab Simulations

The first phase of this study involves the simulation of the MUSIC algorithm in some known computing softwares. The most popular software when comes to digital signal processing is the Matlab developed by Mathworks Inc.

Using Matlab, a powerful computing software, the steps in MUSIC algorithm will be simulated first before it will be finally transferred to its hardware implementation. This will ensure that the chosen algorithm is really feasible for actual hardware implementation. Input parameters such as noise level, number of antenna elements, number of snapshots, spacing between elements and direction of arrival will be varied to observe the robustness of the algorithm. Figure 3.2 shows an example of an output in simulation.

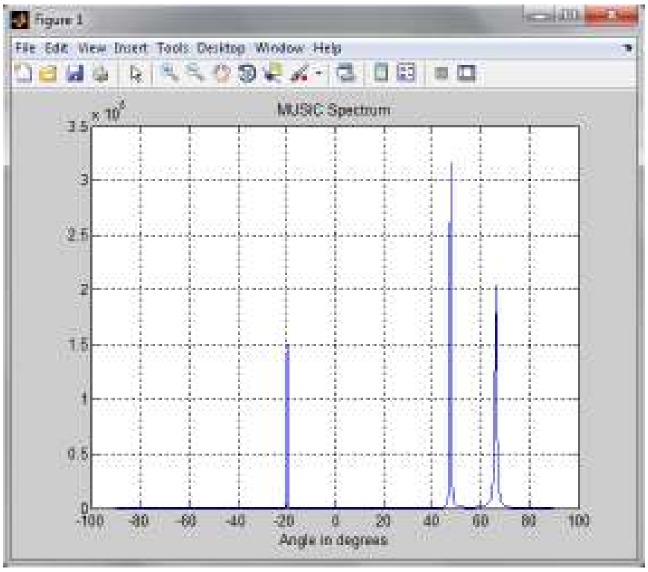

Figure 3.2. DOA simulation using Matlab.

3.3 Construction of the ADC Circuit

The ADC board will play a significant role in this study. This is where the signal will be processed first. An ADC can be implemented using a sample-and-hold circuit. Figure 3.3 shows an example of a sample-and-hold circuit that can be implemented in this study [52].

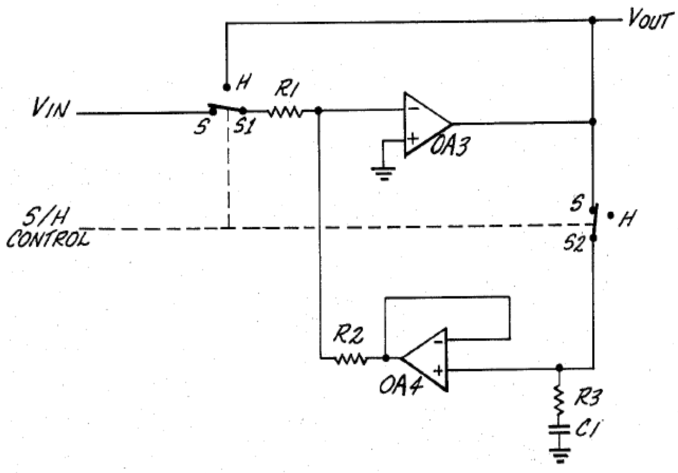

Figure 3.3. A sample-and-hold circuit diagram.

The circuit above consists of two operational amplifiers, OA3 and OA4, which are connected in circuit during the sample and hold modes of the operation. OA3 receives the incoming signal via R1 during the sampling operation which controls the magnitude of an inverted voltage stored on capacitor C1. On the other hand, OA4 senses the stored inverted voltage and applies a feedback voltage to the signal input of OA3 through resistor R2. During the hold operation, OA4 senses the voltage stored on C1 and controls the output voltage through R2 and OA3. Also during the hold mode, the output voltage is fed back via R1 to the input of OA3 receiving the output of OA4. The voltage stored at C1 will only be equal to the input signal voltage if R1 equals R2. The magnitude of the stored voltage will only be greater than the signal input voltage if the ratio of R2 to R1 is greater than one.

3.4 Encoding in the FPGA

In order to implement MUSIC in FPGA, codes used in Matlab should be converted into a language that FPGA can recognize. This is where the role of a hardware description language (HDL) is necessary. For this particular study, Verilog HDL will be employed. Because of its popularity, Verilog became IEEE Standard 1364 in 1995. In year 2002, the IEEE working group released a revised standard known as IEEE 1364-2001.

3.5 Prototype Testing

To show the optimum performance of the algorithm in the FPGA implementation, there are several ways in which the prototype will be tested (refer fig. 3.4 for the testing set-up). The sound source to be used in this study will be the speakers. The performance of the sensor array in estimating for the direction of the source signal will be tested in two types of condition: with and without background noise.

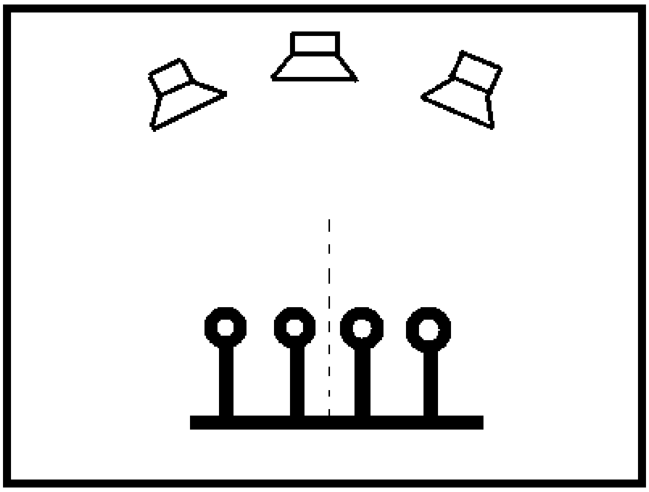

Figure 3.4. Prototype set-up for testing and experimentation.

3.5.1 DOA Estimation Without Background Noise

The first set-up will be using a single source only. One speaker will be placed in ten pre-determined different locations and their corresponding measurements will be taken. The tabulation of the results will be recorded in table 3.1. For the second set-up, an additional sound source will be included and will be positioned also in different locations. Gathered data will be recorded in table 3.2. To further test the accuracy of the array's performance, three sound sources will be considered next which will also be placed in various directions. Results will be put in table 3.3. Lastly, to show that the array's performance will be affected if the sound source will be equal to or greater than the number of antenna elements, four sound sources will be tested. Results will be tabulated in table 3.4.

Table 3.1. DOA estimation for single source.

Single Source	Measured DOA
5°	
10°	
15°	
20°	
25°	
30°	
35°	
40°	
45°	
50°	

Table 3.2. DOA estimation for two sources.

Dual-Source		Measured DOA	
S1	S2	S1	S2
5°	80°		
10°	75°		
15°	70°		
20°	65°		
25°	60°		
30°	55°		
35°	50°		

40°	45°		
45°	40°		
50°	35°		

Table 3.3. DOA estimation for three sources.

Triple Source			Measured DOA		
S1	S2	S3	S1	S2	S3
5°	15°	30°			
10°	20°	35°			
15°	25°	40°			
20°	30°	45°			
25°	35°	50°			
30°	40°	55°			
35°	45°	60°			
40°	50°	65°			
45°	55°	70°			
50°	60°	75°			

Table 3.4. DOA estimation for four sources.

Quadruple Source				Measured DOA			
S1	S2	S3	S4	S1	S2	S3	S4
5°	15°	25°	35°				
10°	20°	30°	40°				
15°	25°	35°	45°				
20°	30°	40°	50°				
25°	35°	45°	55°				
30°	40°	50°	60°				
35°	45°	55°	65°				
40°	50°	60°	70°				
45°	55°	65°	75°				
50°	60°	70°	80°				

3.5.2 DOA Estimation With Background Noise

For the condition with background noise, additional sound sources will be included in the set-up but in low power only to see the effect in the array's performance. A pre-determined DOA value for each sound source will be tested in different levels of signal-to-noise ratio (SNR). Data and results will be placed in their respective table below.

Table 3.5. DOA estimation for single source with noise.

Single Source	Power (dB)		SNR	Measured DOA
	Signal	Noise		
30°			15:1	
30°			10:1	
30°			5:1	
30°			1:5	
30°			1:10	
30°			1:15	

Table 3.6. DOA estimation for two sources with noise.

Double Source		Power (dB)		SNR	Measured DOA	
S1	S2	Signal	Noise		S1	S2
30°	45°			15:1		
30°	45°			10:1		
30°	45°			5:1		
30°	45°			1:5		
30°	45°			1:10		
30°	45°			1:15		

Table 3.7. DOA estimation for three sources with noise.

Triple Source			Power (dB)		SNR	Measured DOA		
S1	S2	S3	Signal	Noise		S1	S2	S3
30˚	45˚	60˚			15:1			
30˚	45˚	60˚			10:1			
30˚	45˚	60˚			5:1			
30˚	45˚	60˚			1:5			
30˚	45˚	60˚			1:10			
30˚	45˚	60˚			1:15			

REFERENCES

[1] A. M. Sayeed, E. N. Onggosanusi and B. D. Van Veen, "A canonical space-time characterization of mobile wireless channels," *IEEE Communications Letters*, vol. 3, pp. 94-97, April 1999

[2] cnx.org/content/m12561/latest

[3] W. Zeng and X. Li, "High-resolution multiple wideband and nonstationary source localization with unknown number of sources," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3125-3136, June 2010

[4] A. Nehorai and E. Paldi, "Acoustic vector-sensor array processing," *IEEE Transactions on Signal Processing*," vol. 42, no. 9, pp. 2481-2451, September 1994

[5] M. Hawkes and A. Nehorai, "Acoustic vector-sensor beamforming and capon direction estimation," *IEEE Transactions on Signal Processing*, vol. 46, no. 9, pp. 2291-2304, September 1998

[6] M. Hawkes and A. Nehorai, "Acoustic vector-sensor processing in the presence of a reflecting boundary," *IEEE Transactions on Signal Processing*, vol. 48, no. 11, pp. 2981-2993, November 2000

[7] N. Zou and A. Nehorai, "Circular acoustic vector-sensor array for mode beamforming," *IEEE Transactions on Signal Processing,* vol. 57, no. 8, pp. 3041-3052, August 2009

[8] P. Bergamo, S. Asgari, H. Wang, D. Maniezzo, L. Yip, R. Hudson, et al., "Collaborative sensor networking towards real-time acoustical beamforming in free space and limited reverberance," *IEEE Transactions on Mobile Computing,* vol. 3, no. 3, pp. 211-224, September 2004

[9] S. Shahbazpanahi, S. Valaee and M. H. Bastani, "Distributed source localization using ESPRIT algorithm," *IEEE Transactions on Signal Processing,* vol. 49, no. 10, pp. 2169-2178, October 2001

[10] A. Beck, P. Stoica and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Transactions on Signal Processing,* vol. 56, no. 5, pp. 1770-1778, May 2008

[11] V. Cevher and J. McClellan, "General direction-of-arrival tracking with acoustic nodes," *IEEE Transactions on Signal Processing,* vol. 53, no. 1, pp. 1-12, January 2005

[12] F. Gao, A. Nallanathan and Y. Wang, "Improved MUSIC under the coexistence of both circular and noncircular sources," *IEEE Transactions on Signal Processing,* vol. 56, no. 7, pp. 3033-3048, July 2008

[13] P. Tichavsky, K. T. Wong and M. D. Zoltowski, "Distributed source localization using ESPRIT algorithm," *IEEE Transactions on Signal Processing,* vol. 49, no. 11, pp. 2498-2510, November 2001

[14] K. S. Lee and S. P. Lee, "A real-time audio system for adjusting the sweet spot to the listener's position," *IEEE Transactions on Consumer Electronics,* vol. 56, no. 2, pp. 835-843, May 2010

[15] J. C. Chen, K. Yao and R. E. Hudson, "Source localization and beamforming," *IEEE Signal Processing Magazine,* pp. 30-39, March 2002

[16] P. Stoica and J. Li, "Source localization from range-difference measurements," *IEEE Signal Processing Magazine,* pp. 63-66, November 2006

[17] I. Potamitis and E. Fishler, "Speech activity detection of moving speaker using microphone array," *IEEE Electronics Letters,* vol. 39, no. 16, pp. 1223-1225, August 2003

[18] S. Marano, V. Matta, P. Willett and L. Tong, "Support-based and ML approaches to DOA estimation in a dumb sensor network," *IEEE Transactions on Signal Processing,* vol. 54, no. 4, pp. 1563-1567, April 2006

[19] S. C. Chan and H. H. Chen, "Uniform concentric circular arrays with frequency-invariant characteristics – theory, design, adaptive beamforming and DOA estimation," *IEEE Transactions on Signal Processing,* vol. 55, no. 1, pp. 165-177, January 2007

[20] H. W. Chen and J. W. Zhao, "Wideband MVDR beamforming for acoustic vector sensor linear array," *IEEE Proceedings on Radar Sonar Navigation,* vol. 151, no. 3, pp. 158-162, June 2004

[21] H. Abeida and J. P. Delmas, "MUSIC-like estimation of direction of arrival for noncircular sources," *IEEE Transactions on Signal Processing,* vol. 54, no. 7, pp. 2678-2690, July 2006

[22] J. Dmochowski, J. Benesty and S. Affes, "Direction of arrival estimation using the parameterized spatial correlation matrix," *IEEE Transactions on Audio, Speech and Language Processing,* vol. 15, no. 4, pp. 1327-1339, May 2007

[23] www.comm.utoronto.ca/~rsadve/Notes/DOA.pdf

[24] R. Woods, J. McAllister, Y. Yi and G. Lightbody, *FPGA-Based Implementation of Signal Processing Systems*, West Sussex, United Kingdom: John Wiley and Sons, 2008

[25] S. Kuo, B. Lee and W. Tian. (2006). *Real-time digital signal processing*. England: Wiley

[26] R. Nitzberg. (1999). *Radar signal processing and adaptive systems*. Norwood, MA: Artech House Inc.

[27] S. Amin, R. Azim, S. P. Rahman, F. Habib and A. Hoque, "Estimation of direction of arrival (DOA) using real-time array signal processing and performance analysis," *International Journal of Computer Science and Network Security,* vol. 10, no. 7, pp. 43-57, July 2010

[28] scholar.lib.vt.edu/theses/available/etd-10302002-220938/unrestricted/Thesis.pdf

[29] www.openmusiclabs.com

[30] www.mathworks.com/matlabcentral/fileexchange/28650-direction-of-arrival-estimation-with-music-algorithm

[31] ptolemy.eecs.berkeley.edu/papers/96/dtmf_ict/www/node1.html#SECTION00010000000000000000

[32] P. Tichavsky, K. Wong and M. D. Zoltowski, "Near field/far field azimuth and elevation angle estimation using a single vector hydrophone," *IEEE Transactions on Signal Processing,* vol. 49, no. 11, pp. 2498-2510, November 2001

[33] research.microsoft.com/en-us/projects/Microphone_Array

[34] S. D. ZhangXin, "Digital processing system for digital beamforming antenna," *IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies f or Wireless Communications Proceedings,* pp. 173-175, 2005

[35] Z. Zou, W. Hongyuan and Y. Guowen, "An improved MUSIC algorithm implemented with high-speed parallel optimization for FPGA," *IEEE International Symposium on Antennas, Propagation and EM Theory,* pp. 1-4, 2006

[36] M. Kim, K. Ichige and H. Arai, "Implementation of FPGA based fast DOA estimator using unitary MUSIC algorithm," *IEEE Vehicular Technology Conference,* vol. 1, pp. 213-217, 2003

[37] M. Kim, K. Ichige and H. Arai, "Real-time smart antenna system incorporating FPGA-based fast DOA estimator," *IEEE Vehicular Technology Conference,* vol. 1, pp. 160-164, 2004

[38] B. Yao, H. Li, T. Zhou, B. Chen and H. Yu, "Real-time implementation of multiple sub-array beam space MUSIC based on FPGA and DSP array," *5th IEEE International Symposium on embedded Computing* , pp. 186-191, 2008

[39] WM Zong and JY. Zhang, "Research and FPGA implementation of MUSIC DOA of smart antenna array," *International Conference on Networks Security, Wireless Communications and Trusted Computing* , vol. 2, pp. 189-192, 2009

[40] S. Soda et al., "Positioning system for mobile terminals using microphone array network as an intuitive interface," *Joint Workshop on Hands-Free Speech Communication and Microphone Arrays,* pp. 155-156, 2011

[41] C. Hao and W. Ping, "The high-speed implementation of direction-of-arrival estimation," *IEEE International Conference on Communications, Circuits and System,* vol. 2, pp. 922-925, 2002

[42] X. Xin, Y. Qinghong, S. Zhaolin, L. Yinan and L. Wenli, "The realization of digital beamforming based on FPGA and DSP," *IEEE International Conference on Intelligent System Design and Engineering Application,* vol. 2, pp. 713-716, 2010

[43] M. Jamali, J. Downey, N. Wilikins, C. Rehm and J. Tipping, "Development of an PFGA-based high-speed FFT processor for wideband direction of arrival applications," *Radar Conference* , pp. 1-4, 2009

[44] A. Garcia, A. Mendez and J.L. Gonzales, "Fuzzy system for DOA estimation in mobile communications using an FPGA," *17th International Conference on Electronics, Communication and Computer,* 2007

[45] S. Palnitkar, *Verilog HDL: A Guide to Digital Design and Synthesis*, Mountain View, California: Sun Microsystems Inc., 1996

[46] http://www.comm.utoronto.ca/~rsadve/Notes/DOA.pdf

[47] http://research.microsoft.com/en-us/projects/Microphone_Array/

[48] L. Gupta and R. P. Singh, "Array signal processing: DOA estimation for missing sensors," *World Academy of Science, Engineering and Technology,* vol. 75, pp. 1149-1153, 2001

[49] S. S. Jeon, Y. Wang, Y. Quian and T. Itoh, "A novel smart antenna system implementation for broadband wireless communications," *IEEE Transactions on Antennas and Propagation,* vol. 50, no. 5, pp. 600-606, May 2002

[50] H. Li, "Efficient and flexible architecture for AES," *IEEE Proceedings on Circuits Devices System,* vol. 153, no. 6, pp. 533-538, December 2006

[51] P.Y. Chen, R.D. Chen, Y.P. Chang, L.S. Shieh and H. Malki, "Hardware implementation for genetic algorithm," *IEEE Transactions on Instrumentation and Measurement,* vol. 57, no. 4, pp. 699-705, April 2008

[52] http://www.freepatentsonline.com/4302689.pdf