MASTER THESIS

A Study of Implementation of Digital Signal Processing for Adaptive Array Antenna

Supervisor: Associate Prof. Hiroyuki ARAI
Submitted on Feb. 12, 2002

Division of Electrical And Computer Engineering,
Yokohama National University, Japan

00DD901 Minseok Kim

PREFACE

An adaptive array antenna technology is paid attention to a lot of applications in the next generation wireless mobile communications. This technology can form a beam pattern at an intended direction by applying digital signal processing algorithms to the digitized signals from each antenna element. It has had some barriers in commercial uses due to implementation complexities and impractically high costs, but advances in digital device technologies have led to solve the implementation difficulties due to the nonlinearity of an analog device with a digital signal processing.

This paper describes the development of an evaluation prototype system for adaptive array antenna test bed to realize reasonable cost and high performance. And then it presents the examination of their applications. This paper consists of 3 main parts: first, a design of digital prototype system; second, a beamforming antenna as an application; third, a DOA (Direction Of Arrival) estimation system as the other application.

In Chapter 2 the hardware configuration and the design of the digital prototype board that we developed considering the architecture of software defined radio (SDR) are described. In Chapter 3 and Chapter 4 the examinations of the application implementation using this prototype system are discussed. Chapter 3 presents the beamforming technique by MRC (Maximum Ratio Combining). It recombines the output power at maximal ratio by co-phasing received signals at each element like a phased array antenna. In this chapter, the circuit implementation details and the experimental results are presented. The digital calibration methods of errors in adaptive array antenna system caused by non-ideal antenna pattern, mutual coupling between each element and etc are proposed and the circuit implementations of them are also presented. On the other hand, Chapter 4 deals with the implementation issue of DOA estimation problem. As a matter of fact, it is significant to recognize previously the transmission conditions and the environment of radio wave so as to model a multi-signal propagation efficiently and form beampattern properly. Moreover to form beams at some directions, most of beamforming algorithms need the knowledge of DOAs of incident waves in advance. Generally the DOA algorithm is very complex and has heavy computational load. From this point of view, the high performance DOA estimation system is very useful in the array antenna technology. In Chapter 4 the hardware implementation of MUSIC (MUltiple SIgnal Classification) method, a super-resolution DOA estimation method, is discussed. Especially the EVD (Eigen Value Decomposition) process that takes the largest computational load in the whole system is mainly focused and the circuit configuration, the estimate of circuit scale and the expected performance are examined.

Finally, Chapter 5 summarizes and concludes this paper.

TABLE OF CONTENTS

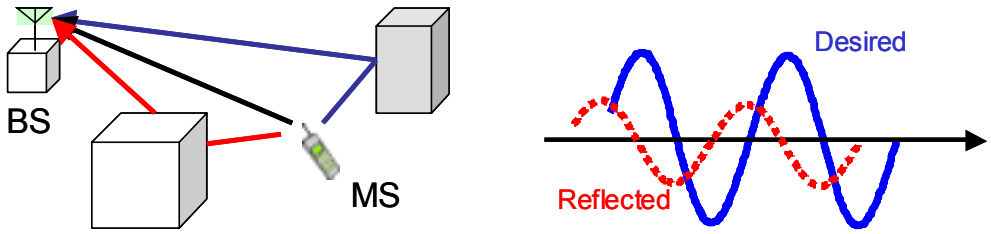
Chapter 1

Introduction

1.1 Background

Wireless Communication technologies have a great progress in recent years and the markets, especially the cellular telephone, have been growing enormously. Moreover the next generation communication services will use higher frequency band area and require more channel capacity and wider bandwidth for a high-speed data communication. As a large increase in channel capacity and high transmission rates for wireless communications, the technologies for the power saving and efficient frequency usability are required.

As a matter of fact in many communication environments there are several serious problems such as a multi-path fading caused by a reflection by any physical structures as Fig. 1-1(a). When passing through multi-path, the signals are delayed and out of phase from the signals through direct-path, as shown in Fig. 1-1(b) that causes the signal strength to be weakened at a receiver and hence receiving quality is also reduced. Generally it is known as “Raleigh Fading”. Wider band and higher transmission rate make it more critical problem in the improvement of the communication quality in the next generation communication. To solve this problem many solutions have been studied. However it is very important to configure receiver and transmitter flexibly in response to the signal environment recognizing spatial profiles as well as temporal ones. For example spatial diversity techniques using composite information from the array to minimize fading and other undesirable effects of multi-path propagation have been studied in many applications. Several techniques such as phased array antenna and diversity antenna using active array configurations can adapt the antenna pattern according to the change of mobile communication environment. However these analog-based techniques have any unavoidable problems that are related to difficult control, absence of individual beam shape control, complex scheme and corresponding heavy equipment and so forth [1].



(a) (b)

Fig. 1-1: (a) Effect of Multi-path from a Mobile User and
(b) Two Out-of-Phase Multi-path Signals

1.2 Survey of Adaptive Antenna

1.2.1 Overview

To meet the requirements of the next generation wireless communications, a system capable of automatically changing the directionality of its radiation patterns (beams) in response to its signal environment must be indispensable. This can noticeably increase the performance characteristics such as capacity and quality of a wireless system. In that regards the other alternative is an adaptive antenna technology. An "Adaptive Antenna" system uses spatially separated antennas called array antenna and processes received signals with a digital signal processor after analog to digital conversion and the name is derived from an "Adaptive Filter" or "Adaptive Filter Signal Processing". This type of antenna that is combined with a digital signal processor is also called by the name of "Smart Antenna" or "Software Antenna" or "Digital Beamforming Antenna (DBF Antenna)" that all mean an intelligent antenna different from a conventional omni-directional antenna only receiving and transmitting signals without any considerations [2]-[4]. On the other hand the term of "Software Defined Radio (SDR)" expected as promising concept of the next generation radio system seems to be used wider sense. An SDR means a set of all target processors required for radio communications as well as antenna technology [5]. An "Adaptive Antenna" is a part of a "Software Define Radio" technology, but the terms are often used as if they have a similar meaning with each other. An "Adaptive Antenna" can be the best explanation of the meaning of "Adaptive Antenna" technology that it is an approach from an antenna in a "Software Define Radio" technology and hence has a close relationship between them in hardware system architecture. This paper makes a rule to use the term "Adaptive

Antenna" from the point of view that it applies digital signal processing technology to the antenna adaptively.

An adaptive antenna can form a beam pattern at an intended direction by applying digital signal processing algorithm with the digitized data from each antenna element. By software algorithm this system at the transmitter is capable of steering the maximum radiation pattern toward a desired mobile and the system at the receiver can spatially separate and reject multi-path fading energy hence higher bit rate services can be provided [6]. Despite of these advantages, there are some obstacles in commercial uses due to implementation complexities and impractically high costs.

1.2.2 Basic Configuration and Principle

Generally an adaptive antenna system consists of a lot of functions including an array antenna and an RF (Radio Frequency) and IF (Intermediate Frequency) circuitry and beamforming network and an adaptive controller. Fig. 1-2 illustrates the basic configuration. An array antenna is plural number of antennas designed to receive or transmit signals using the combined beampattern. There are various physical arrangements of an array such as linear, circular, rectangular, and etc. The structure of an array antenna is determined in consideration of the characteristics and the applications. Most functions of the RF/IF circuitry are frequency downconversion, filtering and amplifying. The beamforming network performs a phase and amplitude control of input signals fed from array antenna, which plays a role in combining beampattern of the array antenna operating as a spatial filter. An adaptive controller determines the optimum weight for beamforming. There are various algorithms for obtaining optimum weight. In an adaptive antenna system, the complex structure, heavy hardware, difficulty to reconstruct and maintenance, etc. with analog technologies come to need the alternative solution. Therefore the configuration using digital technologies has lately considerable attention.

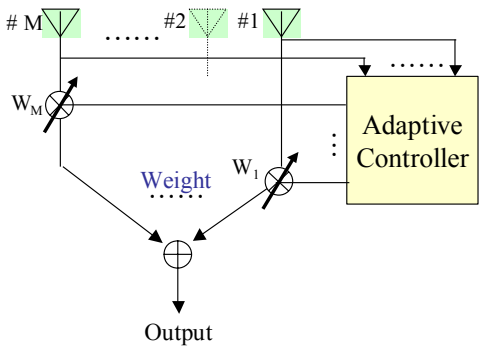


Fig. 1-2: Diagram of Basic Adaptive Antenna System

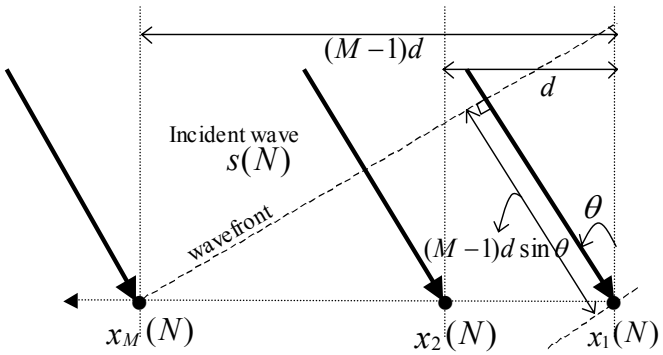


Fig. 1-3: Uniform Linear Array of *M*-Element

The basic structure to explain the principle of adaptive antenna signal processing is illustrated by Fig. 1-3. The array antenna is assumed to be a uniform equidistance linear array of identical and omni-directional M elements and an electromagnetic wave arriving at array antenna is an approximately plane and narrowband signal. Let the angle between wave normal and incident angle θ, the far-field expression of the electrical signal at k-th element at any discrete time N is given by

$$x_k(N) = s_k(N) \cdot \exp(-j\frac{2\pi}{\lambda} dk \sin\theta) \qquad k = 1, 2, \cdots, M \tag{1.1}$$

where $s_k(N)$, λ and θ is the envelop, wavelength and Direction-Of-Arrival (DOA) angle of an incident wave respectively and d is the distance spaced between each antenna. In this equation if $s_k(N)$ is a narrow band signal the temporal delay caused by different path between elements corresponds to the phase difference. The output of array antenna is produced by the inner product (multiply-accumulate operation) of input signals and weight coefficients determined by adaptive algorithms as (1.2).

$$y(N) = \sum_{k=1}^{M} w_k^* \cdot s_k(N) \cdot \exp(-j\frac{2\pi}{\lambda} dk \sin\theta) \tag{1.2}$$

They can be also re-written by vector expression as (1.3)-(1.4).

$$\mathbf{X} = [x_1(N) \quad x_2(N) \quad \cdots \quad x_M(N)]^T \tag{1.3}$$

$$y = \mathbf{W}^H \mathbf{X} \tag{1.4}$$

Basically adaptive antenna technique forms the antenna radiation pattern toward intended direction by digital signal processing. There are two kinds of work. One is beam-steering toward desired direction and the other is null-steering toward undesired interferences, which may be more important function and the original concept of an adaptive antenna. Furthermore historically the

first adaptive antenna is Howells' intermediate frequency (IF) side-lobe canceller for nulling out the effect of one jammer.

On the other hand, to determine the optimum weight, most of the beamforming algorithms such as MMSE (Minimizing Mean Square Error), MSN (Maximizing Signal to Noise ratio), LCMV (Linearly Constrained Minimum Variance Filter), etc. whose solutions are based on solving Wiener-Hopf equation require the information of DOAs of desired signals and interferers in advance. Of course, there are also blind methods in which the information of DOAs is not necessary such as CMA (Constant Modulus Algorithm). As a matter of fact it is significant to recognize previously the transmission conditions and the environment of radio wave. That is to say, the spatial profile such as DOAs of incident signals as well as the temporal profile such as their frequency characteristics is needed. Therefore various techniques of DOA estimation are studied also as a part of adaptive antenna technologies [7].

1.2.3 Performance Improvements

Array signal processing is capable of forming transmit/receive beams towards the desired mobile. At the same time it is possible to place spatial nulls in the direction of undesired interferences called null-steering. This capability can be used to improve the performance of a mobile communication system as follows. The adaptive antenna has a higher gain than a conventional omni-directional antenna. The higher gain can be used to either increase the effective coverage, or to increase the receiver sensitivity. Conversely it can be exploited to reduce transmit power and electromagnetic radiation in the communication network. Multi-path propagation in mobile radio environments leads to inter-symbol-interference (ISI). Using transmit and receive beams that are directed towards the desired mobile reduces the adverse effects of multi-path and ISI. Adaptive antenna transmitters emit less interference by only sending RF power in the desired directions. Furthermore, adaptive antenna receivers can reject interference by looking only in the direction of the desired source. Consequently adaptive antennas are capable of decreasing co-channel-interference (CCI) [8].

1.3 Objective and Structure of Paper

This paper focuses on the implementation issues of an adaptive antenna and examines its applications for practical uses. Historically an adaptive antenna has been mainly used for military applications, but recently practical uses for various applications in wireless communications are expected. However there are many obstacles to realize an adaptive antenna system such as implementation complexity and impractically high cost. The main concept of an adaptive antenna is the automatic or adaptive control of antenna's beampattern by digital signal processing with a software algorithm. An important requirement to realize in current or next generation communications is high-speed realtime processing. But until now the performance of digital devices such as general DSP (Digital Signal Processor) or MPU (Micro Processing Unit) for array signal processing is so poor as to unable to process a large-scale computation and they also consume inefficiently large power to be unsuitable for mobile communications. On the other hand using high performance specific LSI called ASIC (Application Specific Integrated Circuit) bring a low flexibility. A digital device capable of high-speed realtime processing, consuming low power and programmable is required for practical use of an adaptive antenna in wireless communications. In recent year using an FPGA (Field Programmable Gate Array) for the implementation of an adaptive antenna meets the requirements of high performance processing, programmability and low power consumption. It is described in the next section in detail.

This paper examines the practical implementations of an adaptive antenna technique using FPGAs as a digital signal processor. The design and development of a digital prototype system for the evaluation of an adaptive antenna technology are described in Chapter 2 and the application implementations are examined in Chapter 3 and Chapter 4. Beamforming and DOA estimation technique are two typical applications of an adaptive antenna technology. In Chapter 3 the implementation of simple Maximal Ratio Combination (MRC) beamforming antenna and this chapter also discusses a digital calibration techniques of an array antenna system. In Chapter 4 DOA estimation technique using MUSIC (MUltiple SIgnal Classification) algorithm are described and Chapter 5 concludes this paper.

1.4 Implementation Issues on Adaptive Array Signal Processing And FPGA (Field Programmable Gate Array)

There are many requirements needed for implementation of an adaptive antenna technology in the next generation wireless communication system. From the point of view that adaptive antenna is a new concept of antenna combining with digital signal processing unit, the most critical thing can be the performance of a digital signal processor. In other words the high performance for realtime processing of a large-scale computation has been the highest barrier to implement. Considering applications for mobile communication the solution of power consumption problem is also required. In addition reconfigurablity or programmability can improve the communication quality extremely recognizing the communication environment and reconstructing the optimum configuration adaptively, which is a concept of software defined radio.

Particularly to meet the requirement of the real-time processing performance in adaptive antenna techniques takes a complex and high cost array antenna network and a high performance DSP (Digital Signal Processor). Hence in the past they were examined only academically and developed for only special uses such as military radar applications. As the technologies of VLSI have made a great progress nowadays, the processing speed is getting faster and the integration scale is getting larger. SRAM-based FPGA (Field Programmable Gate Array) technology has led to another alternative solution for digital signal processing. In this paper, the system implemented by using an FPGA is introduced. FPGAs in this system play a part as a digital signal processor for digital beamforming or DOA (Direction Of Arrival) estimation functions, which require a large number of MAC (Multiply-ACcumulate) operations and need large-scale parallel processing. FPGAs can meet these requirements.

1.4.1 Basic Descriptions of an FPGA

A programmable logic device (PLD) is loosely defined as a device with configurable logic and flip-flop linked together with programmable interconnect as shown in Fig. 1-4. An FPGA is a kind of programmable logic devices and an array of gates with programmable interconnect and logic functions. It can be reconfigured infinite times after manufacture but generally distinct from PLD by higher logic capacity. An FPGA consists of logic blocks and an interconnection resource to connect the logic blocks. The logic block usually contains lookup tables (LUTs) and flip-flops (FFs) to store data as Fig. 1-5. Input ports are connected to LUT input ports or FF input ports and outputs from LUTs are either connected to output ports of the logic block or connected to FF input

ports. By using multiplexing, various combinations of inputs can be chosen and sequential logics with memory element of FF as well as combinational logics with LUTs are also available [12].

To integrate complex logic circuits distributed to lots of LSI's on board by single or more devices, programmable logic devices have been mainly used. A progress of device integration technology can provide to implement more complex circuits such as large-scale digital signal processing on single FPGA. Furthermore it has many advantages over the other digital signal processing solutions as DSP or MPU and ASIC. The next section describes the advantages in detail.

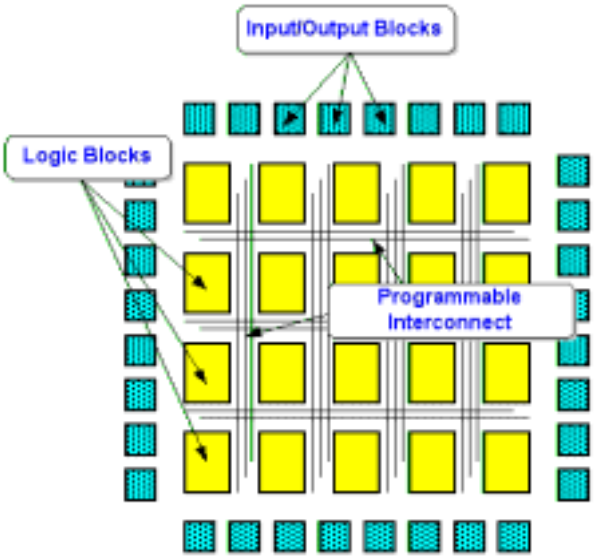


Fig. 1-4: Structure of an FPGA

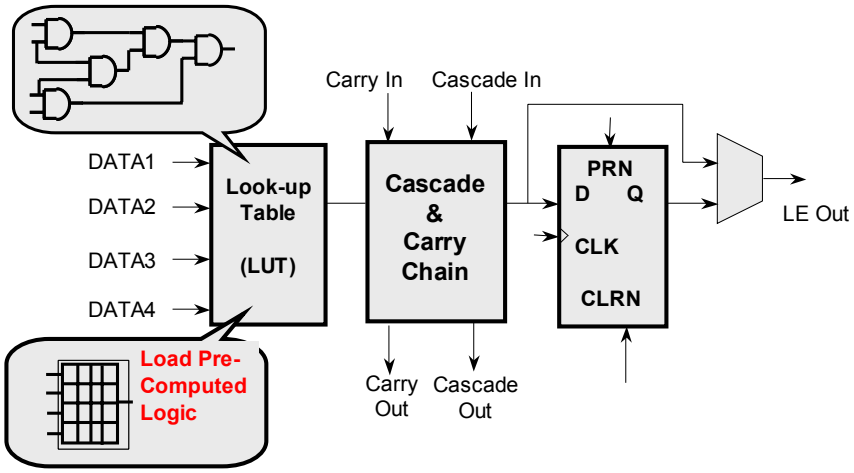


Fig. 1-5: FPGA Logic Block

1.4.2 Digital Signal Processing on FPGAs

As mentioned before, there are a few kinds of digital signal processing solutions for array signal processing techniques. One is using a general purposed processor (DSP or MPU), and the other one is using an ASIC (Application Specific Integrated Circuit). While general purposed processor solutions are very flexible because their architectures are optimized to process a fixed set of instructions but may not be ideally suited to the specific application, ASIC solutions offer the ability to design a custom architecture that is optimized for a particular application. For example, a general purposed conventional DSP has only single multiply-accumulate (MAC) stage, so the computations must be executed sequentially, namely in serial, but whereas an ASIC implementation can have multiple parallel Multiply-ACcumulate (MAC) stages. When comparing the performance of the ASIC versus the general purposed DSP, it becomes apparent that the DSP or MPU offers slow speed but maximum flexibility (programmability) while the ASIC provides high speed with minimal flexibility.

On the other hand, an FPGA combines the versatility of a programmable solution with the performance of dedicated hardware as shown in Fig. 1-6. An FPGA can obtain the true goal of parallel processing executing algorithms with the inherent parallelism due to distributed arithmetic structure while avoiding the instruction fetch and load/store bottlenecks of traditional Von Neumann architectures [11]-[13].

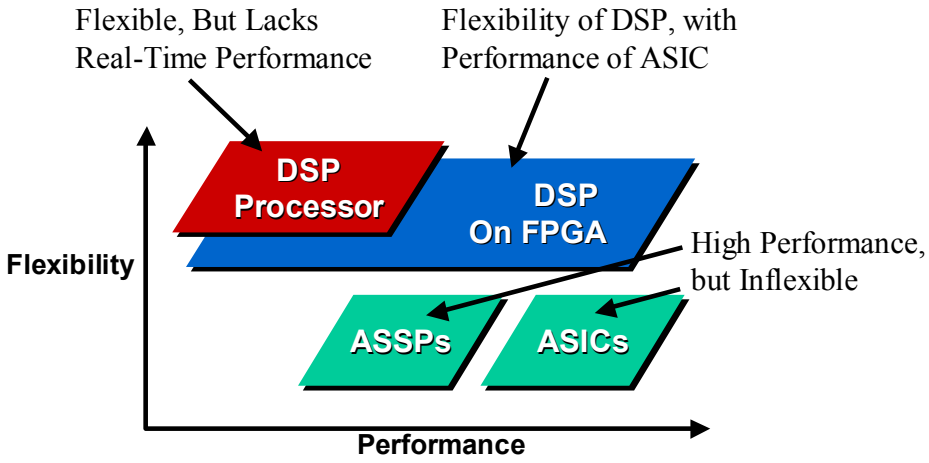


Fig. 1-6: FPGAs offer both Flexibility and Performance

Implementing DSP function in FPGA devices provides the following advantages as Table 1-1 [12]. FPGAs are thought as a key device in implementation of an adaptive antenna or a software defined radio thanks to their high performance, flexibility and reconfigurablity and etc.

TABLE 1-1
A COMPARISON OF FPGA AND DSP PROCESSOR

	FPGA	DSP
Programmable Language	VHDL, Verilog	C, Assembly
Ease of S/W programming	Fairly easy but needs understanding the hardware architecture	Easy
Performance	Very fast if optimized	Speed depends on operating clock speed
Reconfigurablity/ Programmability	SRAM-type FPGAs can be reconfigured infinite times	Re-programmable by changing program
Outperforming Area	Digital Filters, FFT, etc	Sequential processing
Power Consumption	Can be minimized if circuit is optimized	Cannot optimize
Implementation Method of MAC	Parallel and distributive arithmetic	Repeat operation of one or a few MACs
Speed of MAC	Can be fast if a parallel algorithm	Depends on operation clock speed
Parallelism	Can be parallelized for high performance	Usually sequential and cannot be parallelized

Chapter 2

Design of Digital Prototype of Adaptive Antenna Receiver

2.1 Introduction

An adaptive antenna system is a compound technology of many components. As increasing the number of antenna elements, accordingly the system scale gets huger. An adaptive antenna system performs the analog functions of frequency conversion, filtering, gain control and the digital functions of adaptive signal processing, modulation/demodulation and etc after or before A/D or D/A conversion. It is very significant to consider the architecture that provides low costs but meets the performance requirements when designing an adaptive antenna system.

In this chapter, the architectures of an adaptive antenna system are discussed. And the design and development of digital prototype evaluation system on which adaptive antenna techniques are implemented are described. This chapter deals with the implementation of only digital part of IF and baseband stages except analog RF stages. The digital part consists of analog to digital converters (ADCs), FPGAs as a digital signal processor and a CPU for the control of the whole system. This chapter presents the circuit configuration and IF signal processing such as digital downconversion on FPGAs. In addition the relationship to software defined radio architectures through the study of sampling schemes and signal processing on FPGAs is discussed.

2.2 Architectures of Adaptive Antenna System

There are many ways of classifying the architectures of an adaptive antenna system. One of them is the way that how many downconversion stages it has. This way can classify a direct conversion with only single downconversion stage at RF into baseband and a super-heterodyne with a few downconversion stages at RF into baseband via IF. Another is the way that where the ADCs (Analog to Digital Converters) are placed. Generally, the position of ADCs is the most dominant factor of system architecture.

This chapter discusses two types of architectures according to the placement of an ADC. One is a baseband sampling architecture and the other is an IF (Intermediate Frequency) sampling architecture. In addition future trends and the architecture of software defined radio are discussed.

2.2.1 Baseband Sampling Architecture

As shown in Fig. 2-1, it has a few downconversion stages and baseband I/Q signals are derived from mixing the last IF signal with a reference local oscillator. Because the ADC is placed at baseband, the system does not require a high speed and high performance ADC. Usually this system architecture has been used as direct downconversion, double downconversion and triple downconversion according to the number of downconversion stages. Direct conversion has many problems to realize such as a difficulty of building filters to meet the phase and amplitude matching requirements but is attractive from a system downsizing point of view with less RF components. Additional mixers can be added to the direct conversion architecture to improve performance and stability and usually the triple downconversion architecture has been used. This architecture can allow the second IF frequency to be sufficiently low so that a bandpass filter such as a surface acoustic wave (SAW) filter, can be used to define the signal bandwidth. This filter has very low phase and amplitude distortion and hence can provide the high performance. The architectures mentioned above have the analog downconversion stage as the detection of I/Q baseband signals. That causes a few problems as following [14]

- Poor matching between the characteristics of the I/Q signals
- Impairment of I/Q orthogonality
- DC offset
- Spurious noise due to the nonlinearity of the analog components

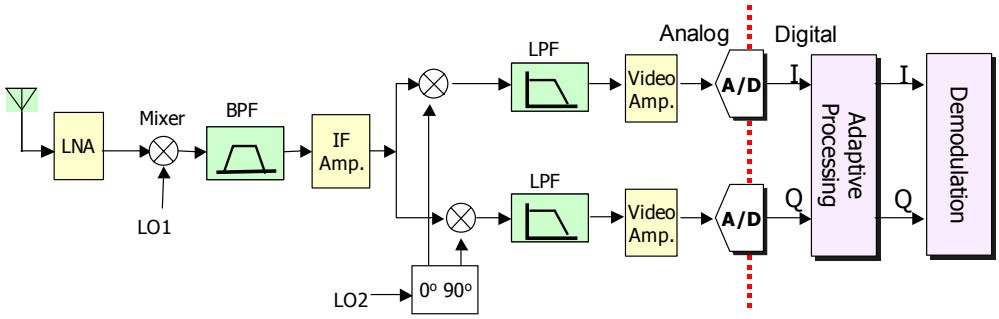


Fig. 2-1: Receiver Architecture with Baseband Sampling

2.2.2 IF (Intermediate Frequency) Sampling Architecture

The alternative conversion architecture is shown in Fig. 2-2. This architecture digitizes signals at IF directly and the complex video signal is generated digitally. In this architecture, the analog mixer and lowpass filters are replaced with digital techniques. Only one high speed ADC provides with decreasing the system circuitry. In addition, the linearity of digital filters solves the matching problems between I/Q signals [14]. A digital downconverter (DDC) is required to perform the coherent detection function. It consists of an NCO (Numerical Controlled Oscillator), a pair of multipliers (mixers in analog sense), lowpass filters and decimations. The decimation reduces data rates which means extracting the narrow baseband signal from the wideband IF input signal. This can allow the digital signal processor to operate at moderate speed. This approach requires the high speed and wide bandwidth ADC and high performance digital multipliers and filters.

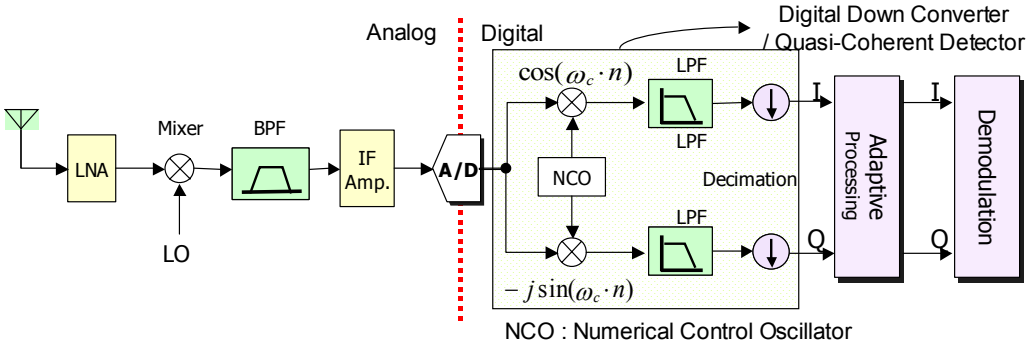


Fig. 2-2: Receiver Architecture with IF Sampling and DDC

2.2.3 Future Trends and Software Defined Radio (SDR)

In the next steps, the ideal architecture as shown in Fig. 2-3 is promising. This approach places an ADC toward antenna as close as possible. Antenna and RF front-end are required to be suitable for receiving wideband signal and ADC must be also able to digitize wide band signal at sampling rates. The other radio functions such as IF, baseband and bit stream processing are carried out using programmable digital processor like a DSP or an FPGA. This is the concept of a software defined radio (SDR). By using programmable or reconfigurable digital signal processing devices multi-mode and multi-band services can be provided and by applying adaptive antenna signal processing it is possible to maximize the system performance and optimize the communication environment.

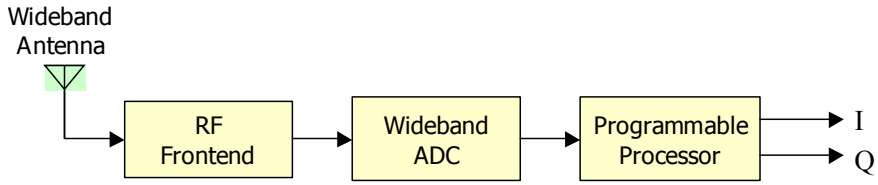


Fig. 2-3: Software Defined Radio Concept

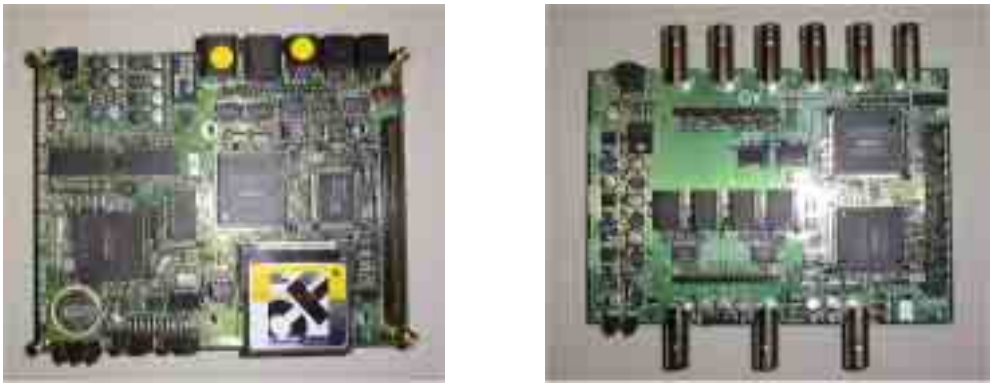
2.3 Development of Prototype System

A few available architectures of an adaptive antenna system were discussed in the previous section. The IF sampling architecture has many advantages over the baseband sampling. The digital downconversion must be the key process to realize a system toward a software defined radio. This section introduces the digital prototype system of IF sampling architecture, which performs I/Q detection digitally on FPGAs.

2.3.1 System Configuration

The digital prototype evaluation system is developed to apply to various adaptive antenna signal processing. It consists of ADC board with ADCs (SPT7938, SPT), and CPU board with a CPU (SH4, HITACH) as a controller. Fig. 2-4 shows the photographs. An ADC board has 2 channels of ADC, buffer memory and 3 FPGAs (total about 300,000 equivalent gates). The CPU board that has CPU, SH4 operating at 200MHz and whose operating system is NetBSD, plays a part in ADC control and as a numerical computation coprocessor. It also offers monitoring

interface and data communication interface between plural ADC boards via Ethernet. A block diagram of the whole system is as shown in Fig. 2-5. ADCs have 12 bit resolution and up to 40 MHz sampling rates. The sampling rates and sampling clock can be controlled by FPGAs on ADC board [15].



(a) (b)

Fig. 2-4: Photographs of (a) ADC Board and (b) CPU Board

As shown in Fig. 2-5 received RF (Radio Frequency) signals at each antenna element are downconverted into IF (Intermediate Frequency). Then this system performs down-conversion and quasi-coherent detection functions by digital signal processing. These features are distinguishing characteristic of a software-defined radio that places ADCs as close to the antenna as possible. In the software radio receiver, the objective is to digitize an entire band and to perform IF processing, baseband, bit stream and all other functions completely in software. However it may be very difficult to implement because such digital radio based IF processing requires such capabilities as high-speed digital filtering, correlation or FFT processing to pass smoothly to the next baseband stages sample by sample in realtime. This prototype system is designed considering the IF processing architecture, key feature of software-defined radio. It can process IF signal up to 10 MHz by 4 times over-sampling in realtime. The examination of sampling scheme is described in the next section. By replacing analog IF downconversion stages to digital signal processing, downsizing of system scale, reducing power consumption and etc can be achieved. After digital downconversion (DDC) / quasi-coherent detection, FPGAs perform adaptive signal processing with down-converted complex baseband I/Q signals. Table 2-1 illustrates the detail characteristics of the system.

This system has only 2 channels of ADCs but multi-channel more than 2 elements can be also configured as Fig. 2-6. It provides sampling clock synchronizing interface with other ADC boards.

They operate at master's sampling clock in common. A data communication bus between them is Ethernet. By this operation the system can extend the number of antenna elements up to 12.

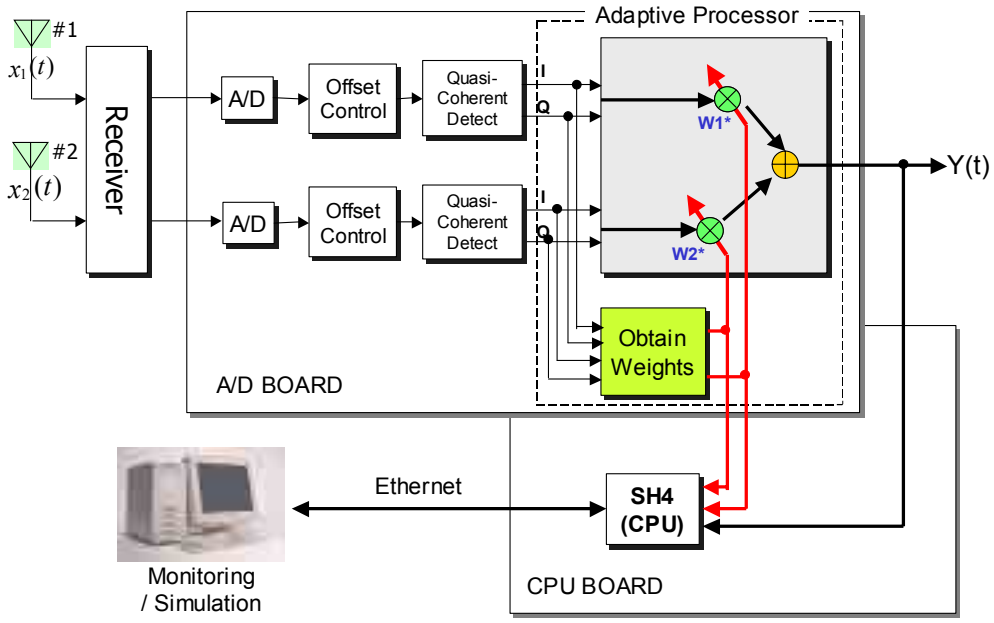
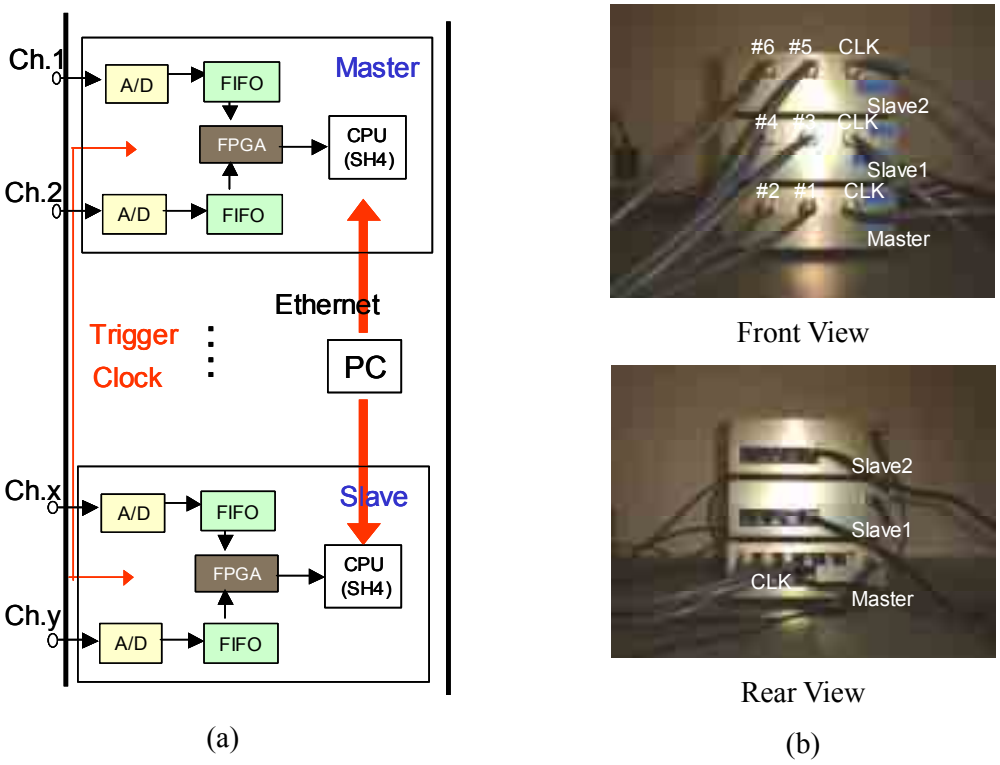
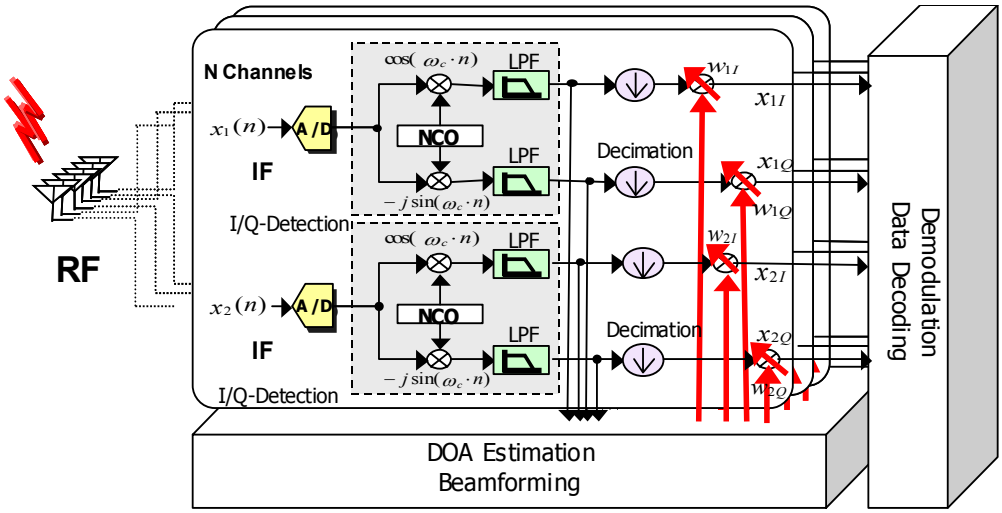


Fig. 2-5: Block diagram of Basic 2-Channel Configuration

Table 2-1
DETAIL SPECIFICATION

ADC	Input Range (IF frequency)	~ 10 MHz (x4 Oversampling)
	Sampling rate	~ 40 MHz
	Resolution	12 bits
	Channel	2 CHs / 1 Board
Digital Downconversion / Quasi-Coherent Detection (on FPGAs)	Input	Passband (12 bits)
	Output	Baseband I / Q (12 bits)
	NCO (Numerical Controlled Oscillator)	Switching Circuit (0, 1, 0, -1)
	LPF (Low Pass Filter)	8 Taps 8 bits Coefficient
FPGAs or CPU	Adaptive Processing	





(c)

Fig. 2-6: (a) Control via Ethernet, (b) 6-element Configuration and (c) Block Diagram of Multi-channel Configuration

2.3.2 Digital Down Conversion (DDC) / Quasi Coherent Detection

Digital Down Conversion / Quasi-coherent detection is implemented by using NCO (Numerical Control Oscillator), mixer and lowpass filters as shown in Fig. 2-7. It consists of sinusoidal signal generator by NCO and 2 FIR (Finite Impulse Response) lowpass filters, where ω_c is angular carrier frequency. If a sampling frequency is 4 times of IF center frequency, NCO and mixer can be easily implemented. In this paper, NCO and mixer are simply implemented by sequence switching circuit (0, 1, 0, -1) as shown in Fig. 2-8. The FIR filters of 8 taps are used.

In digitizing the analog received signals at IF (Intermediate Frequency) only one ADC is required for each antenna element so it can make the system scale by half. In this part, digital signal processing downconvert the sampled bandpass signals from ADC into a complex baseband signal [16]. Bandpass signals can be expressed as a sum of two quadrature components which are $\pi/2$ out of phase. Generally bandpass signal is represented by

$$x(n) = x_I(n)\cos w_c n + x_Q(n)\sin w_c n\,, \tag{2.1}$$

where $x_I(n)$ is the in phase component, $x_Q(n)$ is the quadrature component of the signal $x(n)$ and ω_c is the center frequency of the band pass signal (carrier frequency). The downconversion process shifts the carrier frequency ω_c to baseband. It performs multiplication of the incoming bandpass signal $x(n)$ with the complex phasor $[\cos\omega_c n - \sin\omega_c n]$ and then lowpass filters the result as (2.2).

$$\begin{aligned} x'(n) &= x(n)[\cos w_c n - \sin w_c n] \\ &= \frac{1}{2}[x_I(n) + x_I(n)\cos 2w_c n - j\,x_I(n)\sin 2w_c n + \\ &\quad - j\,x_Q(n) + x_Q(n)\sin 2w_c n + j\,x_Q(n)\cos 2w_c n] \end{aligned} \tag{2.2}$$

This operation accomplishs the desired frequency shift. After lowpass filtering, the second harmonic components are filtered out and the result is the desired complex baseband signal representation of $x(n)$ as (2.3) [17].

$$LPF(x'(n)) = \frac{1}{2}[x_I(n) - j\,x_Q(n)] \tag{2.3}$$

Fig. 2-7 illustrates the block diagram that represents the frequency spectrum as well as this mathematical process. Next sections describe the FPGA implementation details.

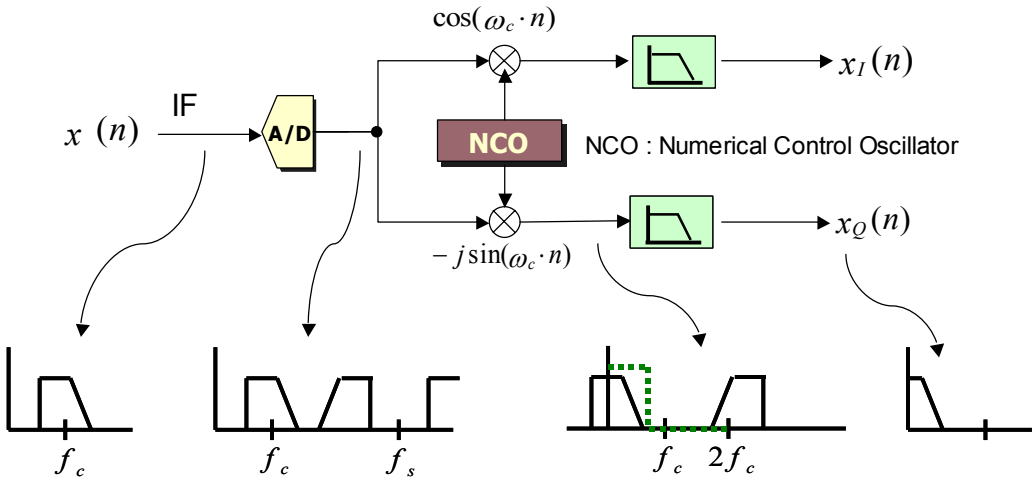


Fig. 2-7: Digital Downconversion / Quasi-Coherent Detection

2.3.3 NCO and Mixer

The downconversion process requires an NCO (Numerical Controlled Oscillator) and a mixer multiplying bandpass signal and digital local sine/cosine signal generated by NCO as shown in Fig. 2-7. There are various methods to implement them. The method generating quadrature signals by DDS (Direct Digital Synthesizer) is usually used. But if clock signal of exactly N times of carrier frequency is achievable, the NCO and mixer are no more than a simple switching circuit as Fig. 2-8. This system performs downconversion function at 4 times of carrier frequency of bandpass signal.

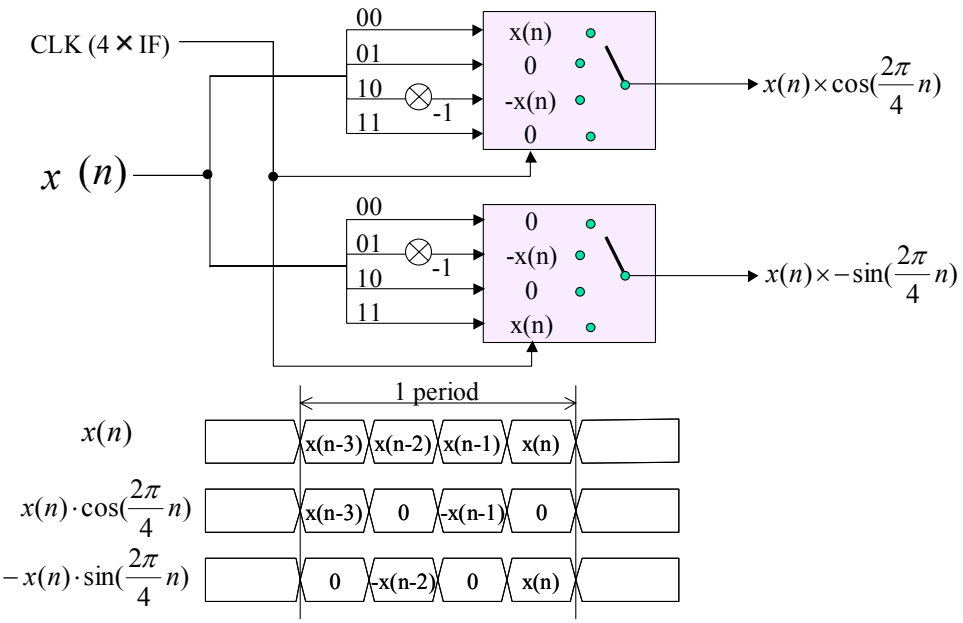


Fig. 2-8: Implementation of Mixer and NCO

2.3.4 Lowpass Filter

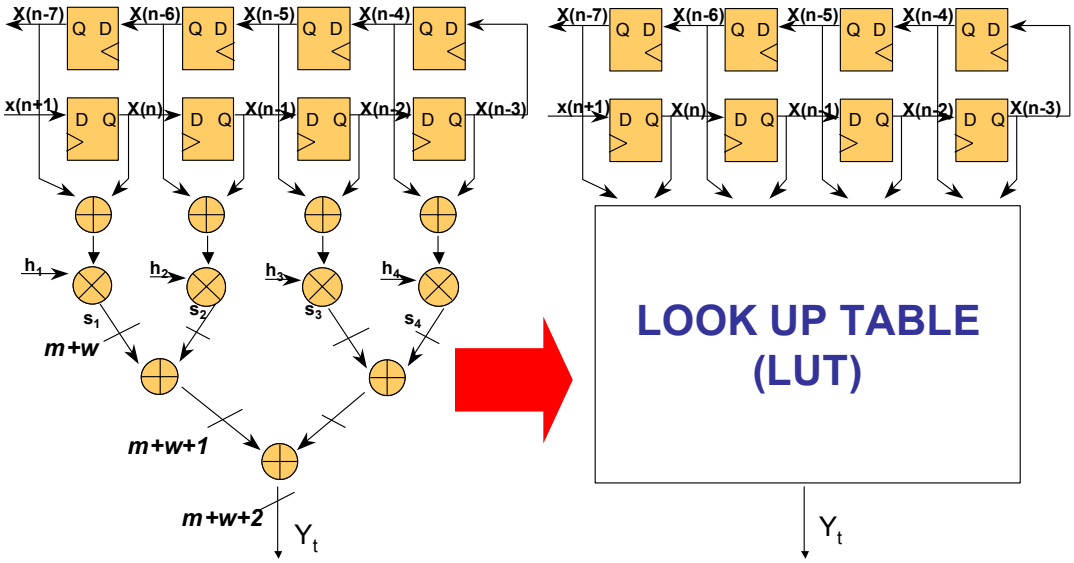
Digital filter implementation on FPGAs has many advantages over the other solution such as general DSP. It is said that a DSP and microprocessor can implement an 8-tap FIR filter at 5 Msps, while an off-the-shelf FIR filter 30 Msps but FPGA can implement the same filter at over 100 Msps because an FPGA is suitable for parallel processing and distributed arithmetic as described in Chapter 1 [18]. Digital downconversion requires two lowpass filters a channel. The inputs are mixed signals of NCO and bandpass signal and the outputs are complex baseband signals of in phase and quadrature. Lowpass filters suppress a second harmonic component and obtain only frequency-shifted signals into baseband. They are FIR (Finite Impulse Response) filters of 8 taps.

In this system, the FIR filter has 12-bit registers arranged in a shift register configuration. The output of each register, called a tap, is represented by $x(n)$, where n is the tap number. Each tap is multiplied by a coefficient $h(n)$ and then all the products are summed. The equation for this filter is

$$y(n) = \sum_{n=1}^{8} h(n) \cdot x(n) \,. \tag{2.4}$$

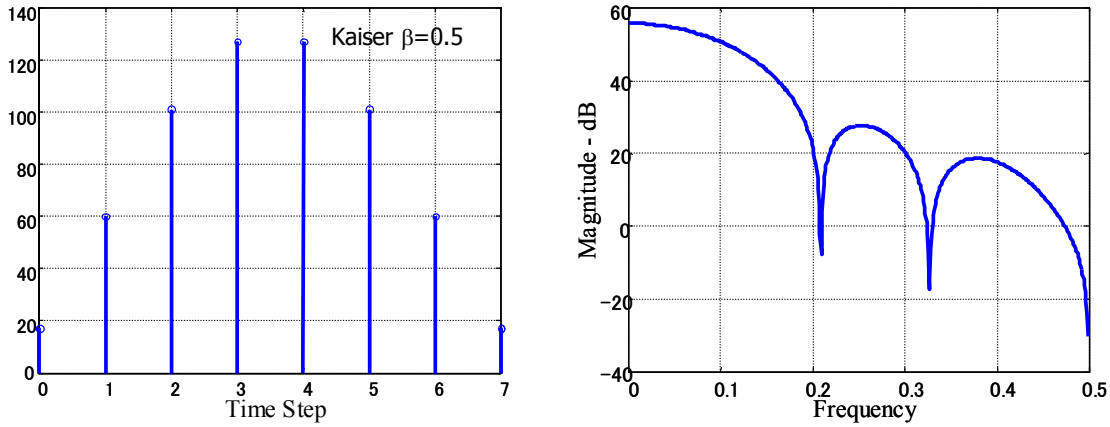
For a linear phase response FIR filter, the coefficients are symmetric around the center values. Taking advantage of the symmetry FIR filter can be reconstructed as Fig. 2-9 (a), which reduces the circuitry required to implement the filter. In addition Fig. 2-9 (a) can be optimized by an FPGA

using look-up tables (LUTs). The multiplication and addition can be performed in parallel using LUTs [18]. The characteristics of FIR filter are shown in Fig. 2-10.



(a) (b)

Fig. 2-9: (a) Conventional FIR filter using Symmetry and
(b) FIR Filter using LUT (Look Up Table)



(a) (b)

Fig. 2-10: (a) Time Domain and (b) Frequency Domain Impulse Response of Lowpass Filter

2.4 Examination of Analog to Digital Sampling Schemes

The system mentioned in previous section provides two schemes of ADC sampling: oversampling including Nyquist sampling and undersampling (also called subsampling or bandpass sampling). This section discusses them and examines the application to our system. This prototype system has the ADC (SPT7938, SPT) which allows a wideband analog input up to 250 MHz and has the sampling rate up to 40 MHz. The diagram of sampling schemes is illustrated in Fig. 2-11.

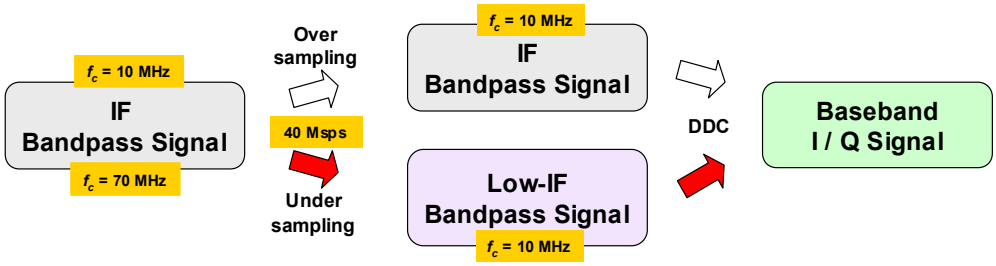


Fig. 2-11: Flows of Each Sampling Scheme

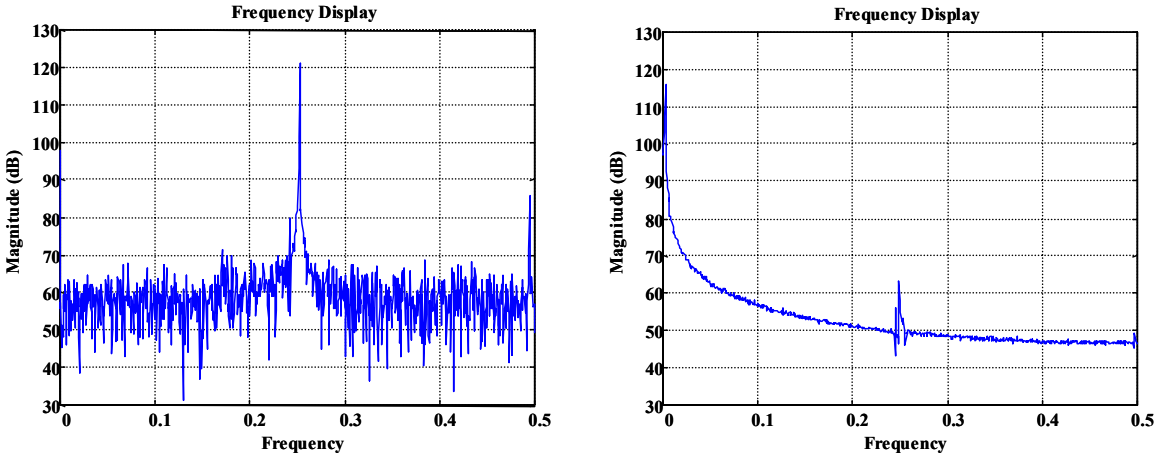
2.4.1 Oversampling Scheme

The sampling rate in analog to digital conversion is determined by Nyquist sampling criterion which specifies the required sampling rate for signal reconstruction as

$$f_s > 2 f_{\max}, \tag{2.5}$$

where f_s is the sampling rate and f_{max} is the maximum frequency of a signal to be digitized. Generally sampling at rates greater than $2f_{max}$ is called oversampling. The advantage of this approach is that the aliase that appears around f_s becomes increasingly separated as the sampling rate f_s is increased beyond $2f_{max}$. By sampling at a higher rate, a simpler anti-aliasing filter with a more moderate transition and less stopband attenuation can be used without any increase in the distortion due to spectrum overlap. Therefore, oversampling can minimize the performance requirements of the anti-aliasing filter whereas faster ADC's are required to digitize relatively low frequency signals.

Using the oversampling scheme this system samples the IF signal (centered at 10 MHz) at 40 Msps and then the DDC (Digital Down Conversion) performs the frequency downconversion into the complex baseband signal. Fig. 2-12 (a) shows the frequency spectrum of sampled IF signal which is sinusoidal signal of 10.1 MHz assuming centered at f_{IF} = 10 MHz. Fig. 2-12 (b) shows the frequency spectrum of downconverted baseband signal.



(a) IF signal (b) Downconverted Baseband signal

Fig. 2-12: Experiment Result of Oversampling (f_{IF} = 10 MHz, f_s = 40 Msps)

2.4.2 Undersampling Scheme

On the other hands, if sampling analog signals at higher frequency area, nonlinear analog functions can be controllable to the extent by digital signal processing so that the concept of a software defined radio can be realized. Less analog components are used and digital signal processing functions replace them. But the performance of an ADC as well as a digital signal processing device is not so high and in addition impractically high cost. In fact it may take Giga-hertz ADC many years to be used usually at reasonable price. The alternative is an undersampling. As a matter of fact, for a bandpass signal, more than two times of the bandwidth of the information can reconstruct the information of the signal. The desired signals will be aliased in band by the undersampling. For the undersampling the sampling rate f_s must satisfy the condition as

$$\frac{2(f_{IF}+B/2)}{k} \le f_s \le \frac{2(f_{IF}+B/2)}{k-1}, \tag{2.6}$$

where f_{IF} is the center frequency of an IF signal and k is an integer number [15][19].

Because conventional ADCs are designed to operate on signals with maximum frequencies up to one-half the sampling rate, conventional ADCs typically are not suitable for bandpass sampling applications where the maximum input frequencies are greater than the sampling rate. In general, performance of ADCs typically degrades with increased input frequency. In addition, stringent requirements on analog bandpass filters are needed to prevent distortion of the desired signal from

strong adjacent channel signals. ADCs at low sampling rates are relatively inexpensive and available and hence this appears promising approach.

Using the undersampling scheme this system samples the IF signal (centered at 70 MHz) at 40 Msps. Fig. 2-13 shows the process of the digital frequency downconversion and the frequency spectrums. In this figure, the aliasings caused by undersampling appear according to *k*'s value in (2.6). The aliasing appearing in-band is the same spectrum of the oversampling. Therefore when signals digitized by undersampling, the frequency is also downconverted into low-IF simultaneously. In this system, IF signal centered at 70 MHz is downconverted into low-IF centered at 10 MHz. The DDC (Digital Down Conversion) performs the frequency downconversion into baseband in the same manner as the oversampling. Fig. 2-14 (a) shows the frequency spectrum of sampled IF signal which is sinusoidal signal of 70.1 MHz assuming centered at f_{IF} = 70 MHz. Fig. 2-14 (b) shows the frequency spectrum of downconverted baseband signal. In the spectrums of Fig. 2-14 more noise and adverse aliasing caused by a spurious components appear than when oversampling. Actually when undersampling, stringent requirements on analog bandpass filters (steep roll-offs) are needed to prevent distortion of the desired signal from strong adjacent channel signals and hence the filter before ADC must be examined significantly.

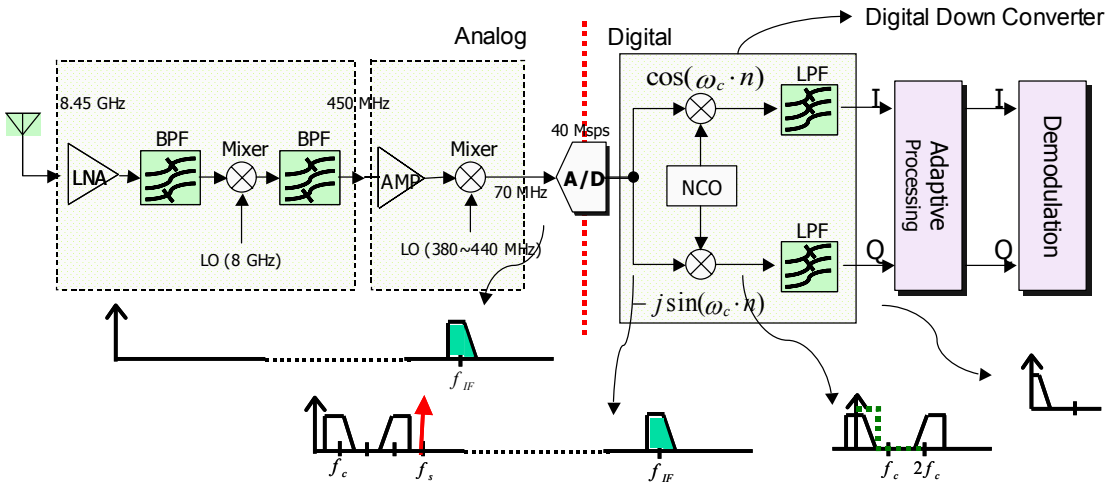
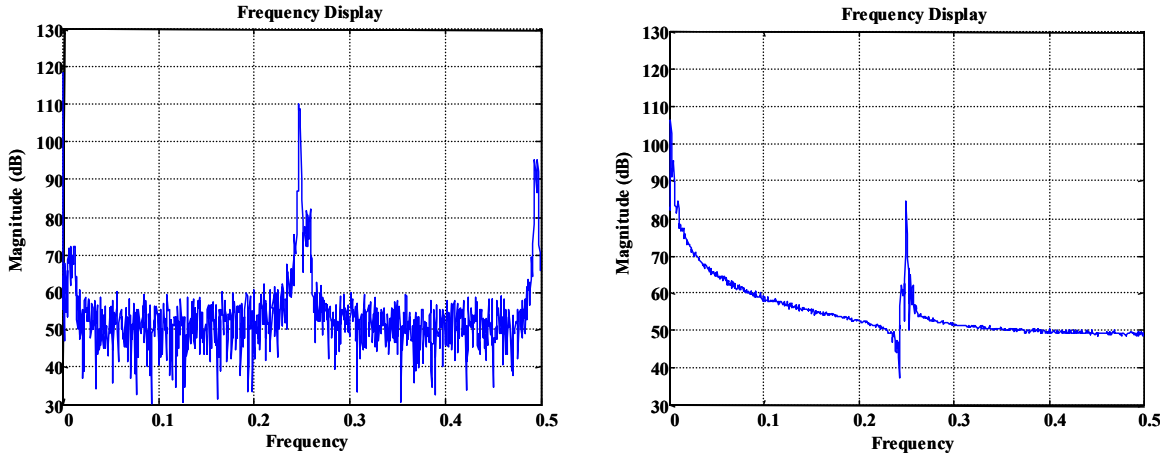


Fig. 2-13: Frequency Downconversion by Undersampling



(a) IF signal (b) Downconverted Baseband signal

Fig. 2-14: Experiment Result of Undersampling (f_{IF} = 70 MHz, f_s = 40 Msps)

2.5 Summary

This chapter introduced the digital prototype system for the evaluation of an adaptive antenna. The architecture was designed based on FPGAs considering IF signal processing and the reconfigurablity of software defined radio (SDR) and the performance of realtime processing required to an adaptive antenna system. The architecture of an adaptive antenna system were discussed and the development processes were described in detail.

Chapter 3

Examination of Applications Implementation: MRC Beamforming Antenna

3.1 Introduction

This chapter describes the simple digital phased array antenna using digital beamforming configuration and introduces its DSP implementation on FPGAs using the prototype system introduced in Chapter 2. This antenna can steer its main beam toward the DOA (Direction of Arrival) of incident signal and track automatically. It incorporates MRC (Maximal-Ratio Combining) beamforming technique that recombines the output power at maximal ratio by co-phasing received signals at each element like phased array antenna. In this paper, for the sake of implementation simplicity and use in mobile terminal, it confines only 2 elements. This system uses FPGAs as digital signal processor to obtain the realtime processing performance of parallel processing.

3.2 2-Element MRC Beamforming Antenna

This paper uses narrowband model for array processing of far-field sources. The incident wave at k-th element can be represented as

$$x_k(n) = A(n) \cdot \exp\left(- j \frac{2\pi}{\lambda} (k-1) d \sin\theta \right), \text{ for } k = 1, 2, \qquad (3.1)$$

where $A(n)$, λ and θ is the envelop, wavelength and DOA angle of an incident wave respectively and d is the distance spaced between each antenna.

After downconversion of the signal (3.1) at each branch, with the complex baseband signal representations (3.1) at each element can be rewritten as

$$\begin{aligned} B_1(n) &= I_1 + jQ_1 = a_1(n)\exp\left[j\left(\Phi_{\text{mod}} + \phi_1\right)\right] \\ B_2(n) &= I_2 + jQ_2 = a_2(n)\exp\left[j\left(\Phi_{\text{mod}} + \phi_2\right)\right], \\ &= a_2(n)\exp\left[j\left(\Phi_{\text{mod}} + \phi_1 + \Delta\phi_{12}\right)\right] \end{aligned} \tag{3.2}$$

where Φ_{mod} is the modulated phase, ϕ_1, ϕ_2 are the phase offset and ϕ_{12} is the phase difference between element 1 and 2. Supposing the receiving power at element 2 is almost same as that of the element 1 and the difference between them is negligible, the sampled data at element 2 can be written as

$$\begin{aligned} B_2(n) &\approx B_1(n)\cdot\exp\left(j\Delta\phi_{12}\right) \\ &\approx B_1(n)\cdot\exp\left(-j\frac{2\pi}{\lambda}d\sin\theta\right). \end{aligned} \tag{3.3}$$

Generally antenna can steer beam toward intended direction of incident wave by using its steering vector as the beamforming optimum weight vector. It is certain that beamforming is achieved by computing the phase difference between elements. The optimum weight can be obtained with only correlations between signals incoming from each element as

$$\begin{aligned} w_1^* &= B_1(n)\cdot B_1^*(n) = \left|B_1(n)\right|^2 \\ w_2^* &= B_1(n)\cdot B_2^*(n) = \left|B_1(n)\right|^2\cdot\exp(j\frac{2\pi}{\lambda}d\sin\theta). \end{aligned} \tag{3.4}$$

It is based on the fact that the correlations between signals incoming from each element and the reference signal (in this case the signal at element 1) represent the steering vector with phase delay caused by DOA of incident signal [1]. Then the optimum weight obtained by correlations between elements are multiplied to input signals as shown in Fig. 3-1.

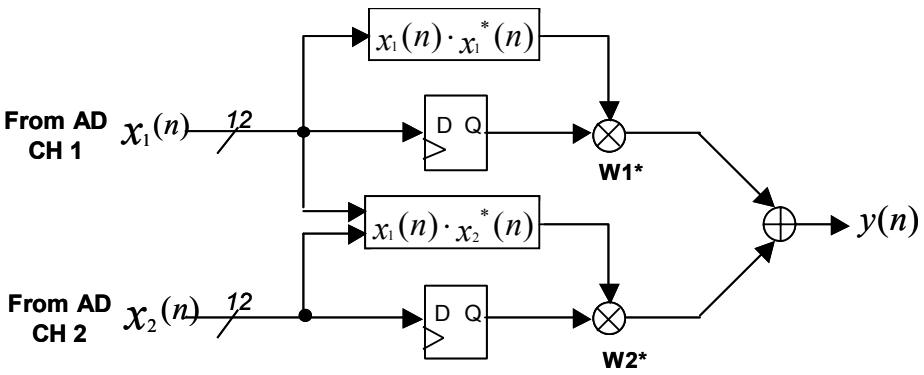


Fig. 3-1: Digital Phased Array Antenna

From which, this system can combine output power at the maximum ratio by multiplying the weights to the input signals as

$$y(n) = \sum_{k=1}^{2} w_k^*(n) \cdot B_k(n) = \mathbf{W}^H \cdot \mathbf{B}\,, \tag{3.5}$$

and conversely it can also find a DOA of incident signal by solving above equation (3.4) of θ.

Fig. 3-2 shows the features that steers its main beam toward the DOA of incident wave and tracks automatically.

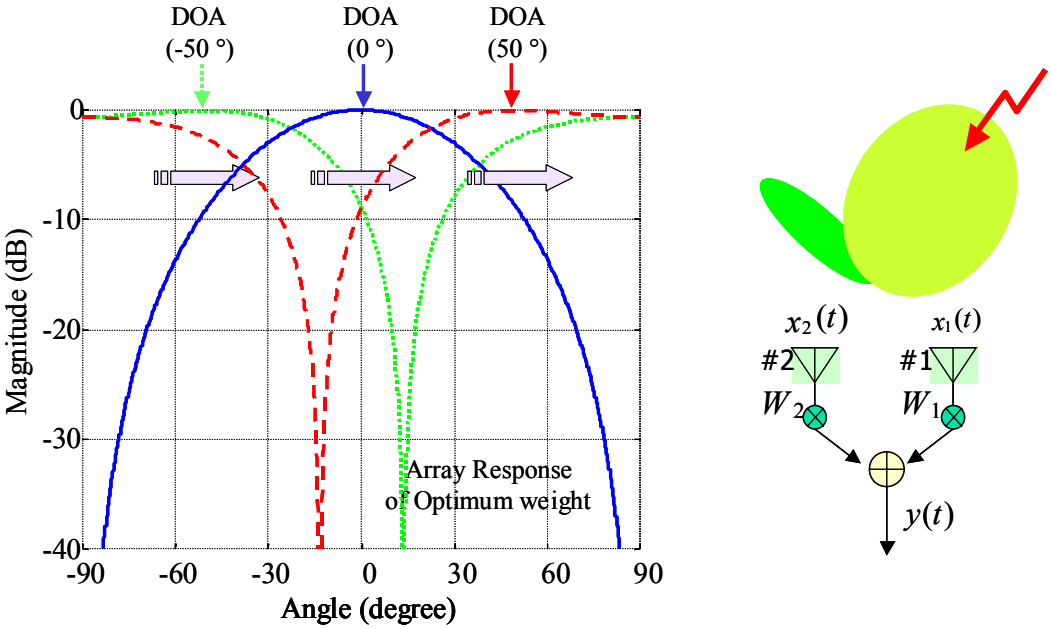


Fig. 3-2: Calculated MRC Beamforming Patterns and Automatic Tracking Feature

3.3 Hardware Implementation

In this section, the hardware implementation of MRC beamforming antenna using the prototype system mentioned in Chapter 2 is described. All digital signal-processing processes are implemented on FPGAs in parallel architecture.

3.3.1 Weight Calculation / MRC (Maximum Ratio Combining) Processor

According to Fig. 3-1 weight calculation processor needs some correlators, multipliers and adders. The real part of correlators can be implemented 6 multipliers and 3 adders as shown in Fig. 3-3 from Eq. (3.4). Multiplying optimum weights to signals from each element the receiving signal can be recombined at maximal ratio in the output power. Fig. 3-3 shows the circuit of producing only real part of outputs and the imaginary part is easily produced in the same circuit structure. All computations on FPGAs are fixed-point operations. The detail I/O data specifications in Hardware design (VHDL description: Very high speed integrated circuit Hardware Description Language) are presented by Table 3-1.

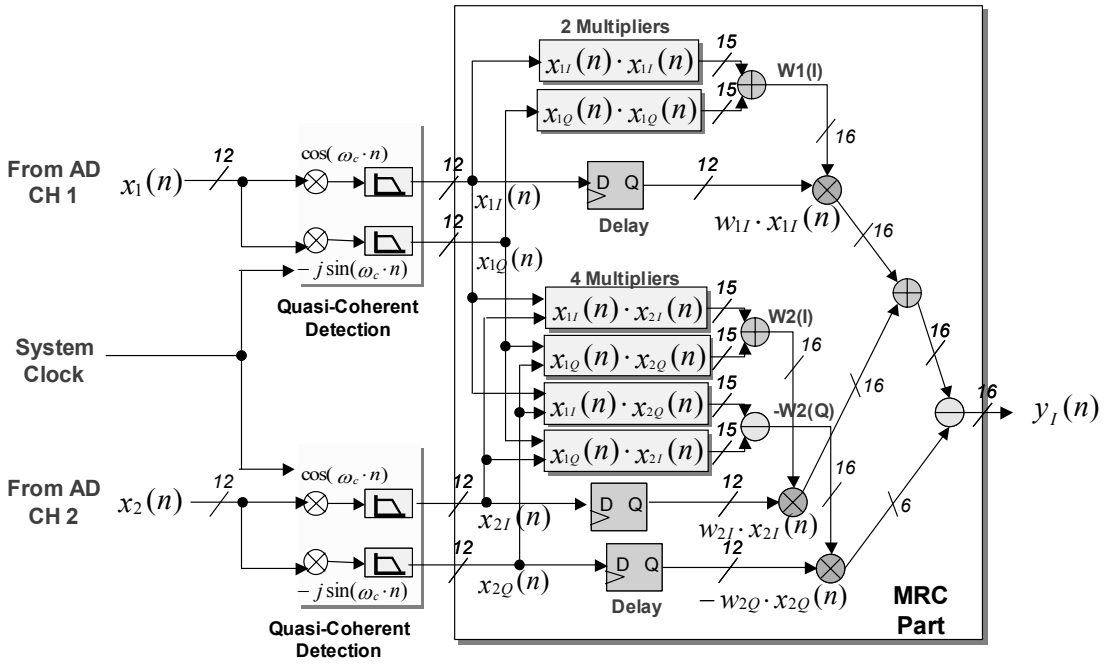


Fig. 3-3: Weight Calculation / MRC Processing Processor (Real-part of outputs)

TABLE 3-1
HARDWARE DESCRIPTION SPECIFICATIONS OF COMPUTATIONS

Module		Function	Specifications	Qty / Remarks
A/D		Output	12 bits unsigned integer	0 ~ 4095
DDC / Quasi-Coherent Detection	NCO	Features	0, 1, 0, -1 sequentially switching	
	LPF	Features	FIR, 8 taps, 8 bits coefficient.	
		Input	12 bits unsigned integer from ADC	1
		Multiplier	Using LUT (Look Up Table) Instead of Multipliers	20 elements (8 bits integer table)
		Adder	12 bits 15 bits 16 bits 20 bits 26 bits	5 3 1 1 1
		Output	MSB 12 bits of 26 bits	2 (I, Q)
Weight Calculation		Input	12 bits signed integer from Quasi-coherent Detector	4 (I, Q of 2 CHs)
		Multiplier	12×12 : scaled 15 bits	6
		Adder	16 bits	3
		Output	16 bits Weights	3 (RE of CH1 IM, RE of CH2)
MRC		Input	12 bits signed integer from Quasi-coherent Detector	3 (I of CH1 I, Q of CH2)
			16 bits signed integer from weight calculation	3 (RE of CH1 IM, RE of CH2)
		Multiplier	16×12 : scaled 15 bits	3
		Adder	16 bits	2
		Output	16 bits Y(n)	1
DOA Calculation		Arctangent Arcsine	32 bits floating point calculation by CPU	

3.3.2 Circuit Design / Logic Synthesis (Scale of Circuit)

The result of logic synthesis of the hardware descriptions written in VHDL (Very high speed integrated circuit Hardware Description Language), a standard language in H/W description of ASIC or FPGA, is shown in Table 3-2. In this result, it can be expected that the scale of circuit needed in implementation of this system is reasonable size to be covered with only one medium scale FPGA in recent technology. If it is used practically, the beam steering function can provides the high quality communication.

TABLE 3-2
RESULTS OF LOGIC SYNTHESIS

Module		Equivalent Gates
DDC (Q-DET)	NCO	0, 1, 0, -1 sequentially switching
	LPF	About 5,931 Gates (8 tap, FIR)
Weight / MRC		About 3,8428 Gates
Total Gates		1 NCO, 2 Q-Detectors, 1 MRC = 5,931×2×2 + 3,8428 Gates = around 6,2152 Gates

3.4 Experimental Results

This section introduces the experimental results. This experiment is DOA estimation with obtained optimum weight. MRC beamforming function steers beam toward DOA of incident signal. Conversely the DOA angle can be obtained from the MRC optimum weight.

From (3.4) the phase difference Θ between signals can be obtained by the arctangent of the imaginary part to real part ratio of optimum weight.

$$\Theta = \frac{2\pi}{\lambda} d \sin\theta = \tan^{-1}\left(\frac{\mathrm{Im}(w_2^*)}{\mathrm{Re}(w_2^*)}\right) \tag{3.6}$$

And the DOA θ can also computed by solving as

$$\theta = \sin^{-1}\left\{\frac{\lambda}{2\pi d}\cdot\tan^{-1}\left(\frac{\mathrm{Im}(w_2^*)}{\mathrm{Re}(w_2^*)}\right)\right\}. \tag{3.7}$$

From (3.7) the DOA of incident signal can be found by MRC optimum weight because MRC steers the main beam toward the direction of incident wave. This operation confirms the beamsteering function of MRC processor.

The experimental configuration is shown as Fig. 3-4. Experiments were performed in a radio anechoic chamber to validate the functionality of the design. Array antennas of 2 elements are omni-directional and spaced by $\lambda/2$ between each other. The RF frequency is 8.45 GHz and received RF signals are downconverted into IF (1 MHz) in the RF receiver. Sampling rates and resolution of ADCs is 4 MHz and 12 bits respectively. This experiment was performed supposing that there is only 1 incident wave and decimation factor is 1. All digital signal processing stages such as DDC, weight calculation and maximal ratio combining are performed on FPGAs. The detail experimental parameters are illustrated in Table 3-3.

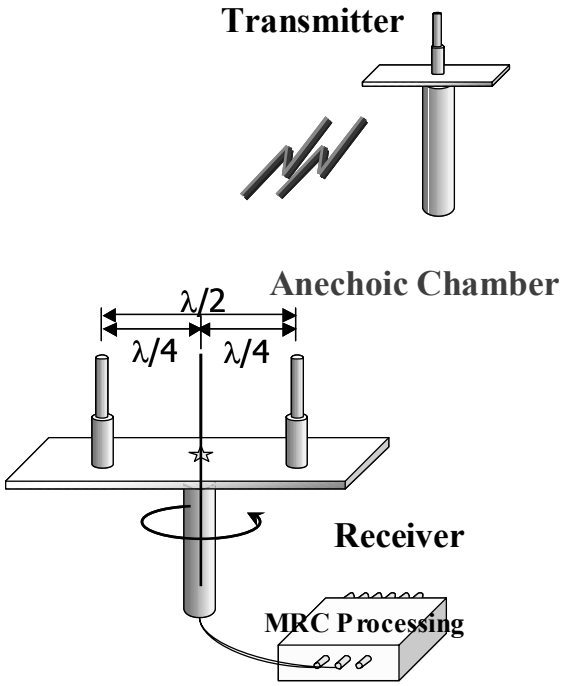


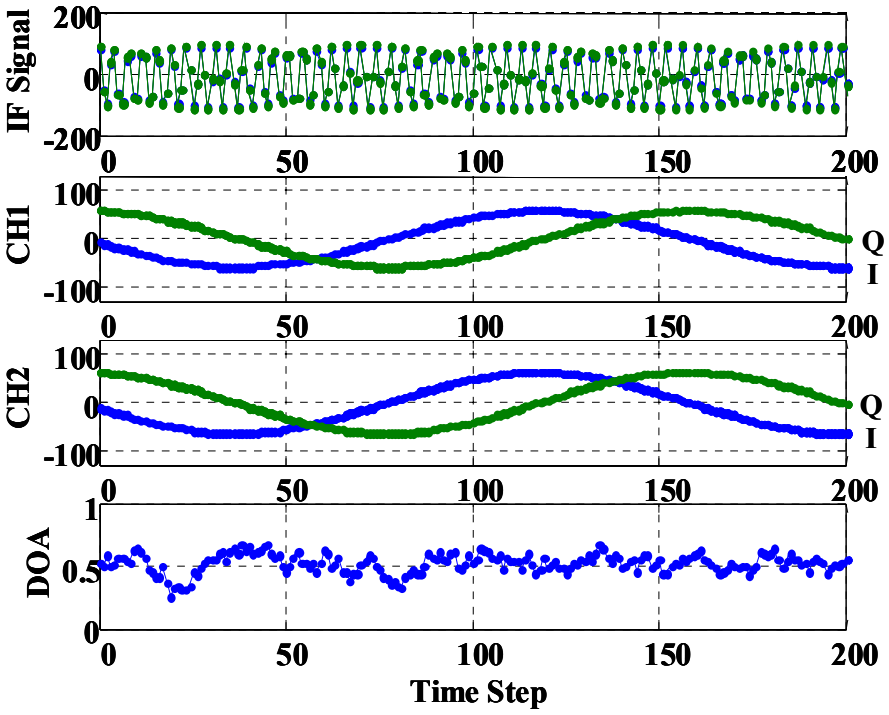
Fig. 3-4: Experimental Configuration

TABLE 3-3
EXPERIMENTAL PARAMETERS

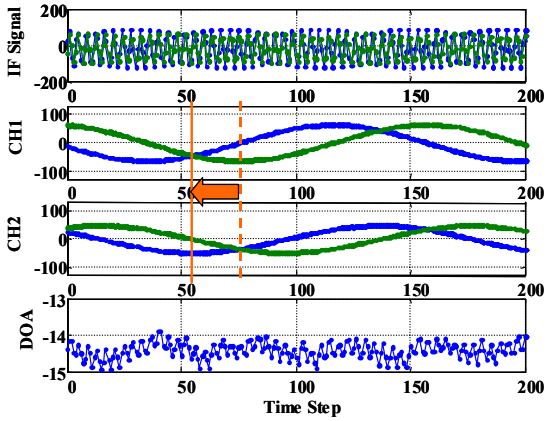
Antennas	2 elements spaced by $\lambda/2$
RF Frequency	8.45 GHz
RF D/C	Analog down-conversion by Receiver
IF Frequency	1 MHz
IF D/C	Digital Down-Conversion
Sampling Rates	4 MHz (×4 Oversampling)
Detection method	Quasi-Coherent Detection
Baseband Freq.	25 KHz
Modulation	CW (unmodulated signal)

3.4.1 DOA Estimation and DOA Tracking

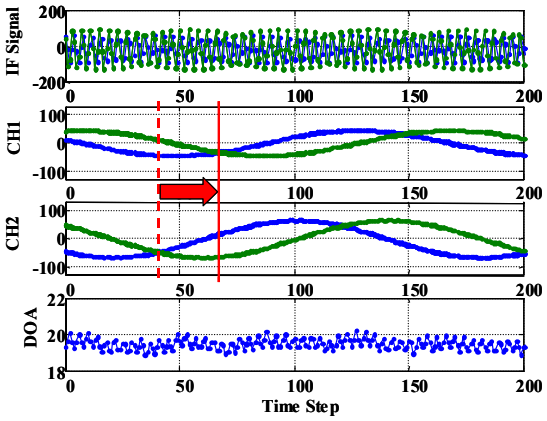
The experimental results at some degrees are shown in Fig. 3-5. IF signals centered at 1 MHz of each branch are digitized at sampling rates of 4 MHz. The second and third figures in Fig. 3-5 illustrate downconverted complex baseband signals at each channel branch. The DOAs were computed offline from the optimum weights produced by FPGAs. By passing through LPF or averaging optimum weights during any period we can get the estimation results. Fig. 3-5 (a) is the result of 0°. It confirms that two channels are co-phased though there seems a small amount of error. (b), (c) and (d) of Fig. 3-5 are the results of ±20°, ±50° and ±60° respectively. They show the system works well enough to steer main beam toward DOA. But (e), (f) and (g) of Fig. 3-5, the results of ±70°, ±80° and ±90° respectively, fail in beamsteering. In this result, it is certain that in the range between about -60° and 60° the beamsteering function of this system works well but as getting closer to 90° beyond 60° the estimation error becomes greater and the beamsteering comes to failure. The successful results from 0° to 60° at every 10° are drawn in Fig. 3-6 and Fig. 3-7 shows the beampatterns calculated by the MRC optimum weight at that degrees.



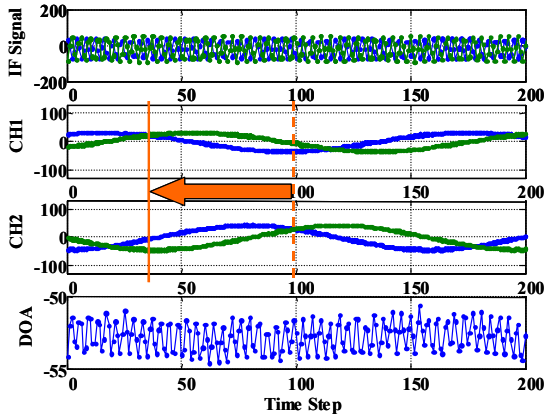
(a) IF Sampling, Frequency DDC / Q-Detection and DOA Estimation at 0 degree



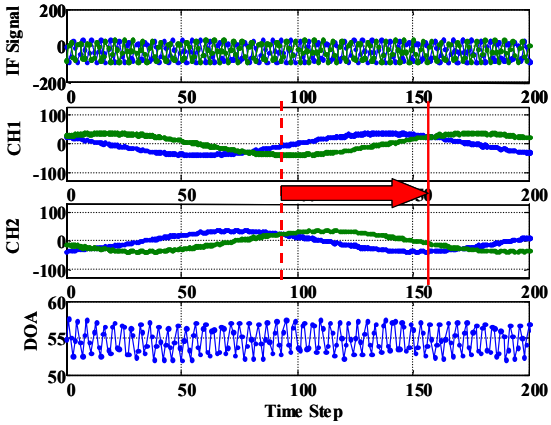
(b-1) -20 degrees



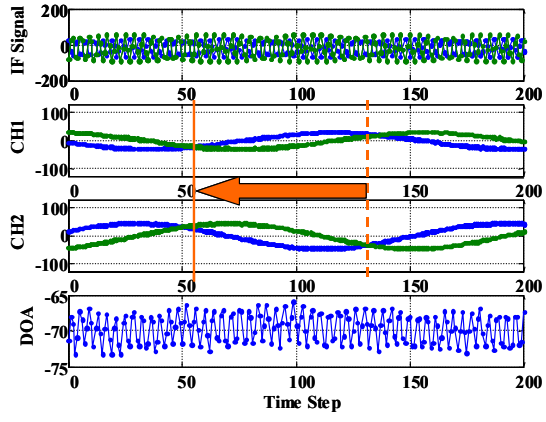
(b-2) +20 degrees



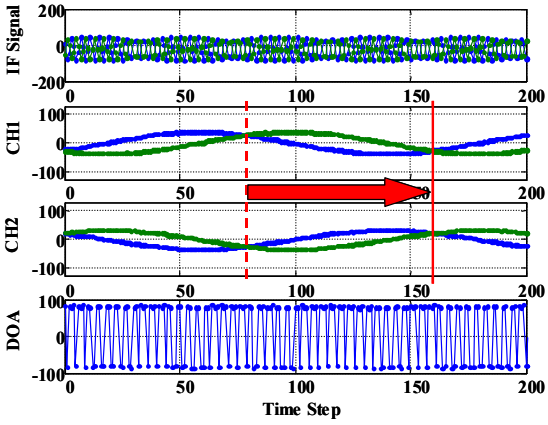
(c-1) -50 degrees



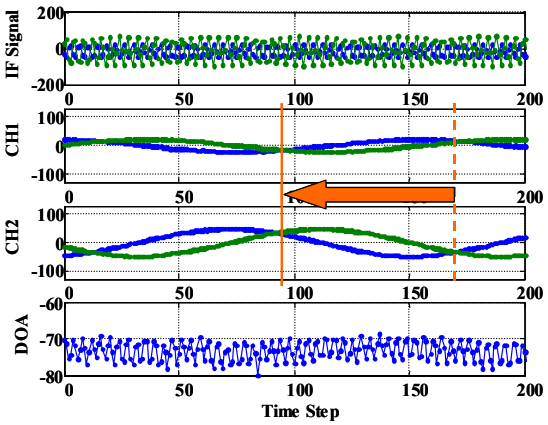
(c-2) +50 degrees



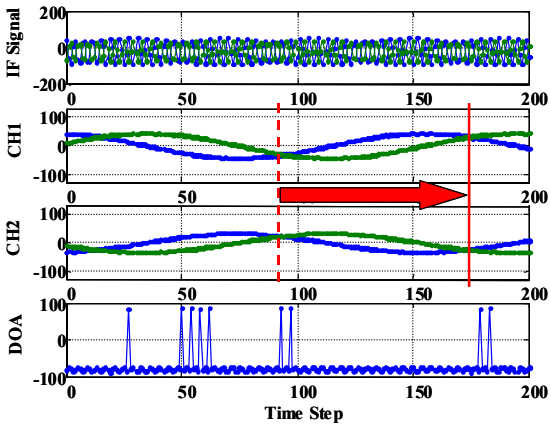
(d-1) -60 degrees



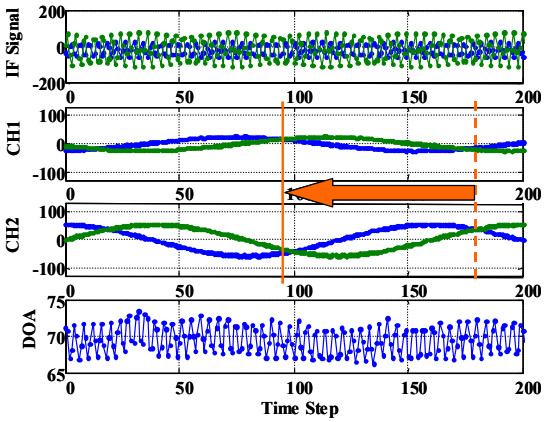
(d-2) +60 degrees (failure)



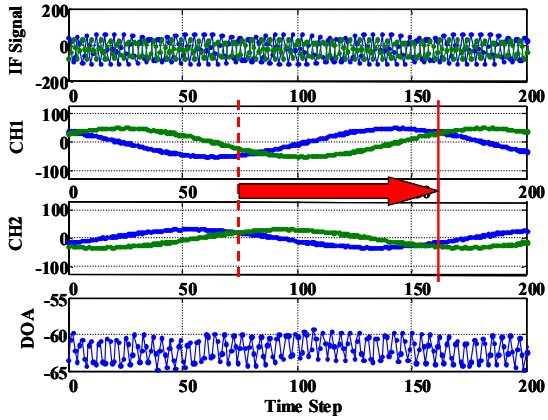
(e-1) -70 degrees



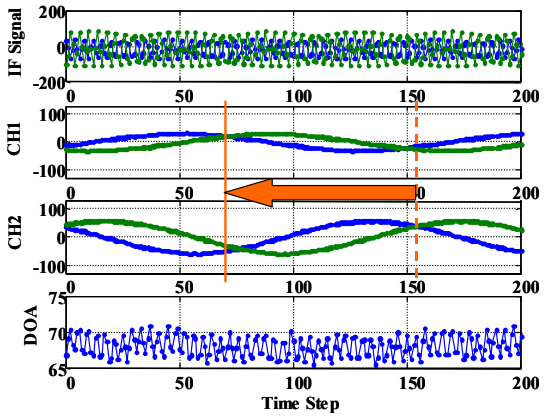
(e-2) +70 degrees (failure)



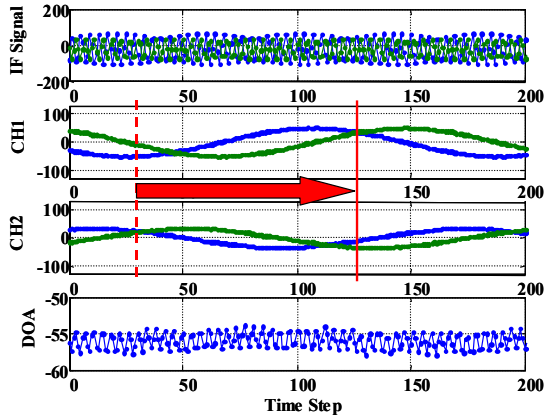
(f-1) -80 degrees (failure)



(f-2) +80 degrees (failure)



(g-1) -90 degrees (failure)



(g-2) +90 degrees (failure)

Fig. 3-5: IF Sampling, Frequency DDC / Q-Detection and DOA Estimation

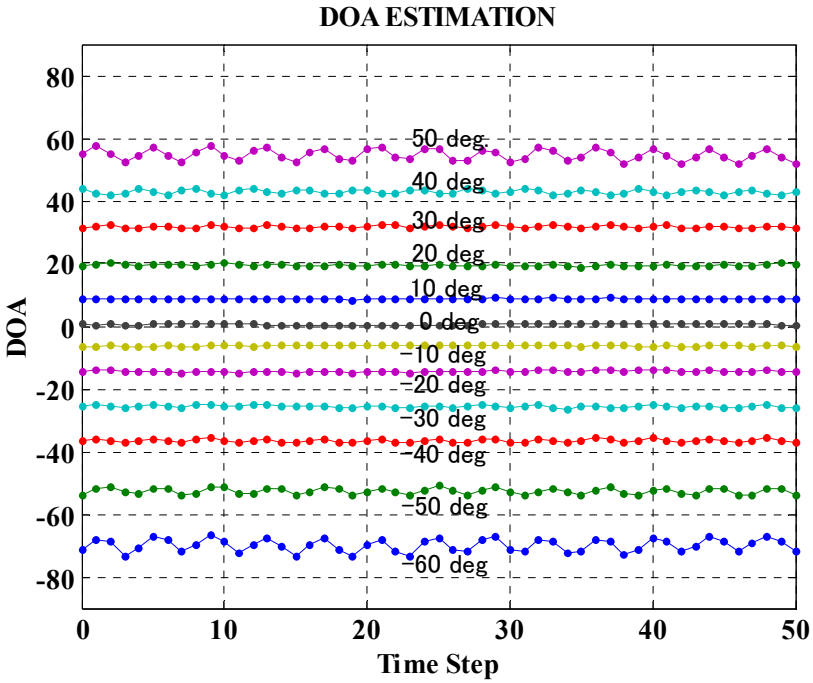


Fig. 3-6: DOA Estimation Results at every 10°

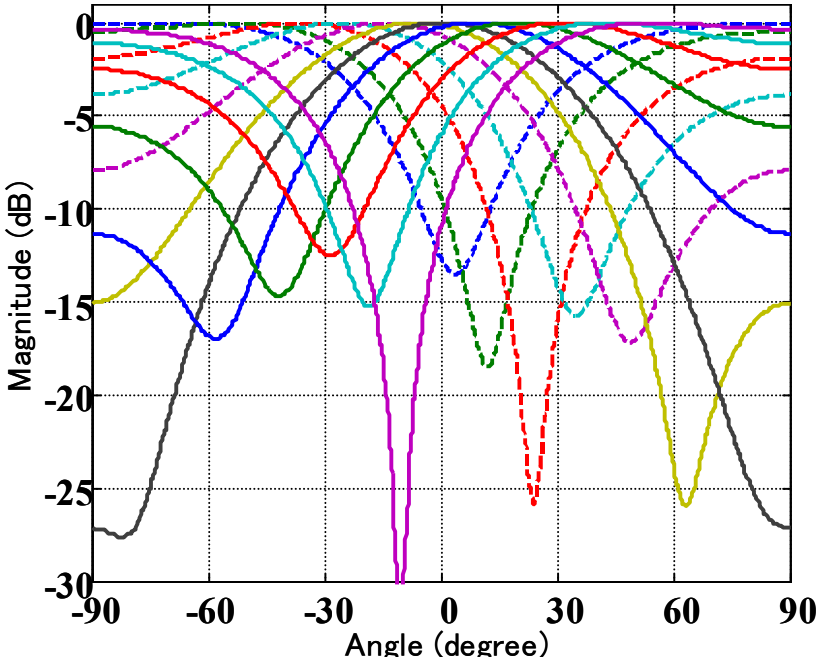


Fig. 3-7: Beampattern formed by MRC optimum weight when the DOA is at every 10°

This system can track automatically sample by sample in realtime. Fig. 3-8 shows well arcsine curve characteristic despite moving at linear speed due to the fact that DOA is the arcsine function of phase difference between signals from (3.6),(3.7). Exceeding $\pm 60°$ the estimation values become unstable because the phase difference between signals becomes to greater than $\lambda/2$ and computation results are unbelievable in the vicinity of 60°.

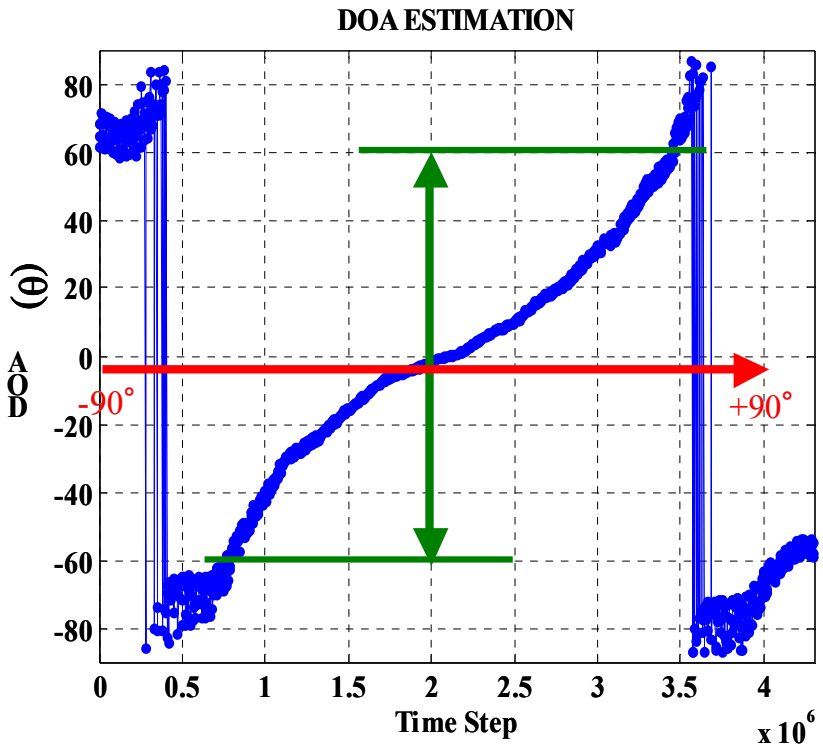


Fig. 3-8: DOA Tracking Feature (moving linearly from -90° to 90°)

3.4.2 Discussion

According to the previous subsections, only in the range of DOA between -60° and +60° the beamsteering function of the system operates well. The error of the phase difference between signals becomes larger causes the unbelievable results. There may be many error factors. Fig. 3-9 shows an antenna placement error. If used high frequency, this error is very critical. For more accurate in wider range of DOA, the change of antenna pattern caused by mutual coupling must be also considered. And actually antennas have not ideal omni-directional radiation pattern and are not identical. Therefore the compensation of errors must be taken into account.

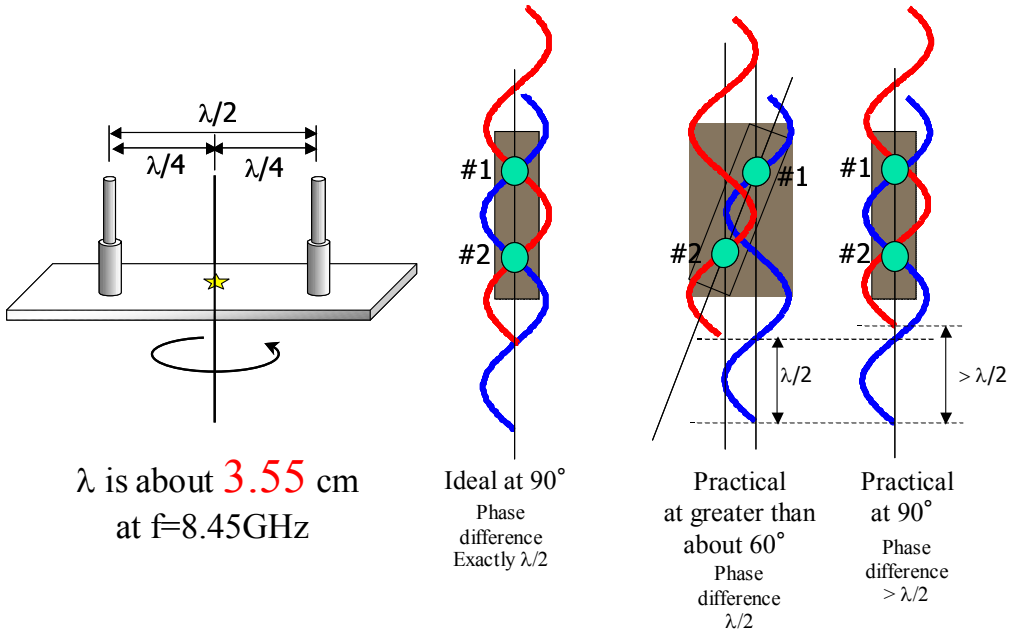


Fig. 3-9: Errors Caused by Antenna Placement

3.5 Examination of Digital Calibration of Array Antenna

Generally there are many error factors of use adaptive antenna array in actual field such as antenna placement error, mutual coupling, nonlinearity of RF devices and etc. To use adaptive array antenna system in actual field, the basic calibration or compensation stages or processes must be required. First of all, the fact that antenna pattern is not ideal and distorts the amplitude and phase of input signal. Generally, this antenna response is a function of DOA angle and each antenna has a different radiation pattern. So the complete calibration requires the information of antenna manifold in advance and needs an additional huge circuitry. Digital approach has many advantages over the calibration using nonlinear analog devices dominated by the environment like temperature.

This section introduces digital approaches to calibration of array receiver. They operate not for all DOAs but the only initial calibration at specified DOA between elements. They play a role in the adjustment to a desired phase and amplitude between signals at any antenna elements. This section proposes two useful digital signal processing techniques easy to implement on an FPGA. One is the adjustment by control of the amplitude and the phase of an NCO (Numerical Controlled Oscillator) that act as mixer in analog sense. The other is using MRC (Maximum Ratio Combining) optimum weight. In this chapter, the basic ideas and a few advantages / disadvantages of the two techniques are described.

3.5.1 Adjustment by NCO Control

The complex baseband signal is obtained by lowpass filtering IF (Intermediate Frequency) signal mixed with or multiplied by NCO as mentioned in section 2.3.2 of Chapter 2. The signal $\tilde{x}(n)$ multiplied by NCO is given by

$$\tilde{x}(n) = x(n) \cdot M \cdot e^{-j\omega \cdot n}, \tag{3.8}$$

where $x(n)$ is an IF signal, M and ω are an amplitude and frequency of NCO respectively. Generally in DDC (Digital Down Conversion) stage, NCO frequency ω is ω_c. From (3.8) it is certain that by using phase shifter and gain controller to NCO at IF DDC stage the phase and amplitude of each signal can be adjusted. Assuming the amplitude of NCO M is 1 in (3.8), the adjusted frequency shifted signal can be obtained by (3.9).

$$\begin{aligned}\hat{x}(n) &= x(n)\cdot e^{-j\omega_c\cdot n}\cdot A e^{-j\phi} \\ &= x(n)\cdot A e^{-j\{\omega_c\cdot n+\phi\}} \\ &= \tilde{x}(n)\cdot A e^{-j\phi}\end{aligned}, \tag{3.9}$$

where A and ϕ are a scalar amplitude and phase angle respectively. Fig. 3-10 shows the block diagram of this operation. Actually an FPGA implement NCO using LUTs of sine/cosine pre-computed value. For example, assuming the IF input signals by

$$\begin{aligned}x_1(n) &= \sin\{(\omega_c+\omega_0)n\} \\ x_2(n) &= \frac{1}{2}\cdot\sin\left\{(\omega_c+\omega_0)n-\frac{\pi}{3}\right\}, \\ x_3(n) &= \frac{1}{4}\cdot\sin\left\{(\omega_c+\omega_0)n+\frac{\pi}{6}\right\}\end{aligned} \tag{3.10}$$

phase shift and amplitude control achieves the adjustment of signals $\hat{x}_k(n)$ as

$$\begin{aligned}\hat{x}_1(n) &= x_1(n)\cdot e^{-j\omega_c\cdot n}\cdot A_1 e^{-j\phi_1} = x_1(n)\cdot A_1 e^{-j\{\omega_c\cdot n+\phi_1\}} = \tilde{x}_1(n)\cdot A_1 e^{-j\phi_1} \\ \hat{x}_2(n) &= x_2(n)\cdot e^{-j\omega_c\cdot n}\cdot A_2 e^{-j\phi_2} = x_2(n)\cdot A_2 e^{-j\{\omega_c\cdot n+\phi_2\}} = \tilde{x}_2(n)\cdot A_2 e^{-j\phi_2}, \\ \hat{x}_3(n) &= x_3(n)\cdot e^{-j\omega_c\cdot n}\cdot A_3 e^{-j\phi_3} = x_3(n)\cdot A_3 e^{-j\{\omega_c\cdot n+\phi_3\}} = \tilde{x}_3(n)\cdot A_3 e^{-j\phi_3}\end{aligned} \tag{3.11}$$

where A_1, A_2, A_3 and ϕ_1, ϕ_2, ϕ_3 are an amplitude and a phase to be adjusted. They must be precomputed. Fig. 3-11 shows the computer simulation results.

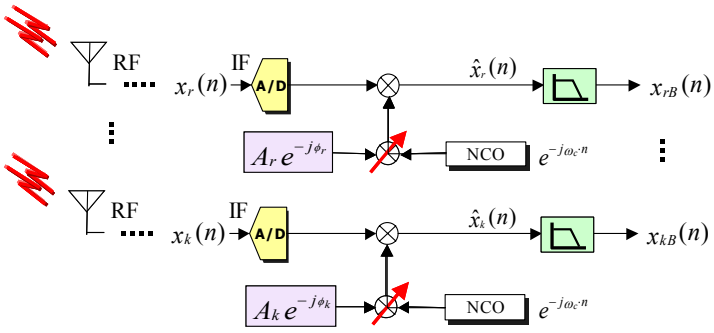
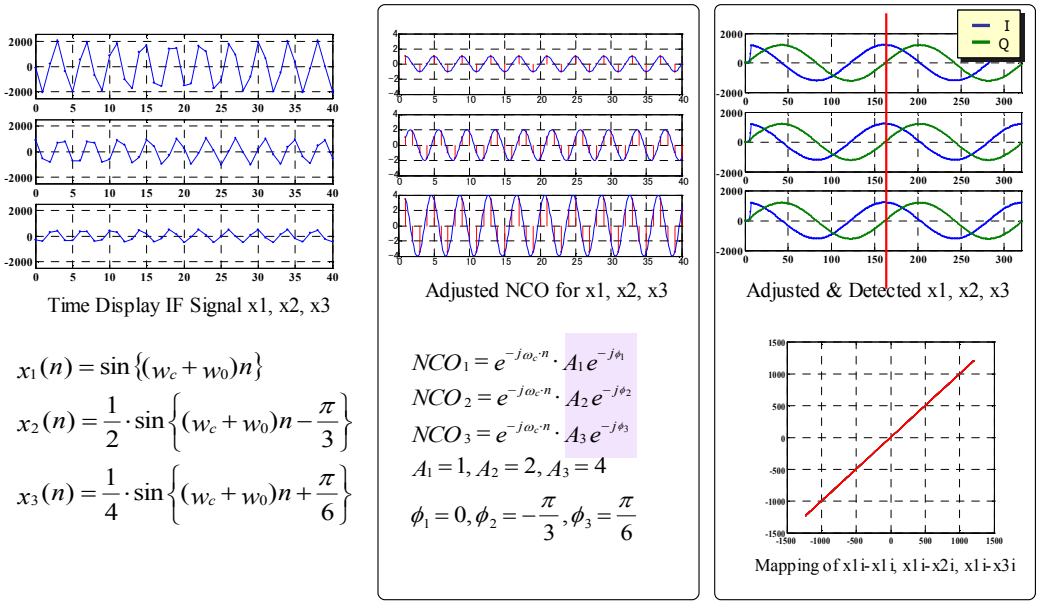


Fig. 3-10: Block Diagram of Adjustment by NCO Control



$$x_1(n) = \sin\{(w_c + w_0)n\}$$
$$x_2(n) = \frac{1}{2}\cdot\sin\left\{(w_c + w_0)n - \frac{\pi}{3}\right\}$$
$$x_3(n) = \frac{1}{4}\cdot\sin\left\{(w_c + w_0)n + \frac{\pi}{6}\right\}$$

$$NCO_1 = e^{-j\omega_c\cdot n}\cdot A_1 e^{-j\phi_1}$$
$$NCO_2 = e^{-j\omega_c\cdot n}\cdot A_2 e^{-j\phi_2}$$
$$NCO_3 = e^{-j\omega_c\cdot n}\cdot A_3 e^{-j\phi_3}$$
$$A_1 = 1, A_2 = 2, A_3 = 4$$
$$\phi_1 = 0, \phi_2 = -\frac{\pi}{3}, \phi_3 = \frac{\pi}{6}$$

Fig. 3-11: Computer Simulation Result of Adjustment by NCO Control

3.5.2 Adjustment by Maximal Ratio Combining (MRC)

If uses narrowband model for array processing of far-field sources as described in previous sections of this chapter, the baseband signals at *r*-th element as a reference signal and at *k*-th element can be represented as

$$\begin{aligned} x_{rB}(n) &= s(n)\cdot A(\theta)\, e^{-j\phi(\theta)} \\ x_{kB}(n) &= s(n)\cdot e^{-j\frac{2\pi}{\lambda}d\sin\theta}\cdot B(\theta)\, e^{-j\zeta(\theta)} \end{aligned}, \qquad (3.12)$$

where $s(n)$ is the baseband complex temporal amplitude, $A(\theta)$, $B(\theta)$ and $\phi(\theta)$, $\zeta(\theta)$ are scalar amplitudes and phase angles respectively which are an antenna response function of θ. λ and θ are the wavelength and the DOA of an incident signal respectively and d is the distance spaced between elements. The MRC optimum weight can be obtained with correlations between signals coming from each element as

$$\begin{aligned} r_{rr}^{\ *} &= x_{rB}(n)\cdot x_{rB}^{\ *}(n) = \left|s(n)\cdot A(\theta)\right|^2 \\ r_{kk}^{\ *} &= x_{kB}(n)\cdot x_{kB}^{\ *}(n) = \left|s(n)\cdot B(\theta)\right|^2 \\ r_{rk}^{\ *} &= x_{rB}(n)\cdot x_{kB}^{\ *}(n) = \left|s(n)\right|^2\cdot e^{j\frac{2\pi}{\lambda}d\sin\theta}\cdot A(\theta)B(\theta)\cdot e^{-j\{\phi(\theta)-\zeta(\theta)\}} \end{aligned}, \qquad (3.13)$$

The MRC optimum weight adjusts the baseband signals to the same amplitude and phase by (3.14) with one another. (3.14) can be applied if the receiving powers of input signals has not the same level by multiplying the scaling factor with each other, which is different from (3.4).

$$\begin{aligned} x'_{rB}(n) &= r^*_{rr} \cdot r^*_{kk} \cdot x_{rB}(n) = w^*_r \cdot x_{rB}(n) \\ x'_{kB}(n) &= r^*_{rk} \cdot r^*_{rr} \cdot x_{kB}(n) = w^*_k \cdot x_{kB}(n) \end{aligned}, \tag{3.14}$$

(3.14) needs only multiply operations, and hence it is easy to implement on an FPGA. It can be incorporated in pre-stage of adaptive signal processing as a vector rotation adjustment process as shown in Fig. 3-12. This calibration process must already know the exact DOA of transmitting calibration signal and it can be the simplest case that the DOA is 0°. Once this initializing process is performed, the resulting rotation vectors are stored in memory (in FPGAs) and are used from the next operation steps.

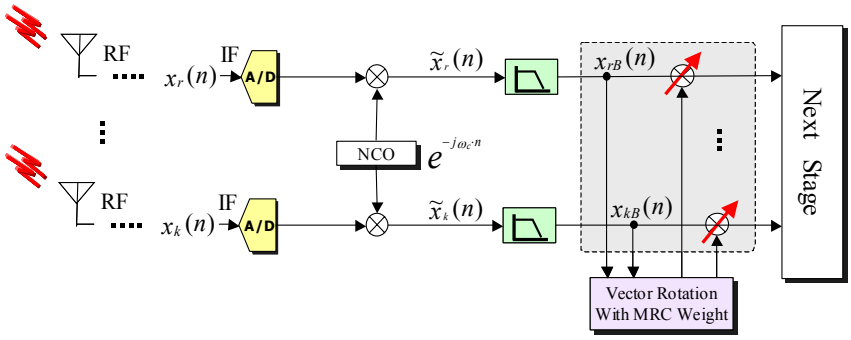
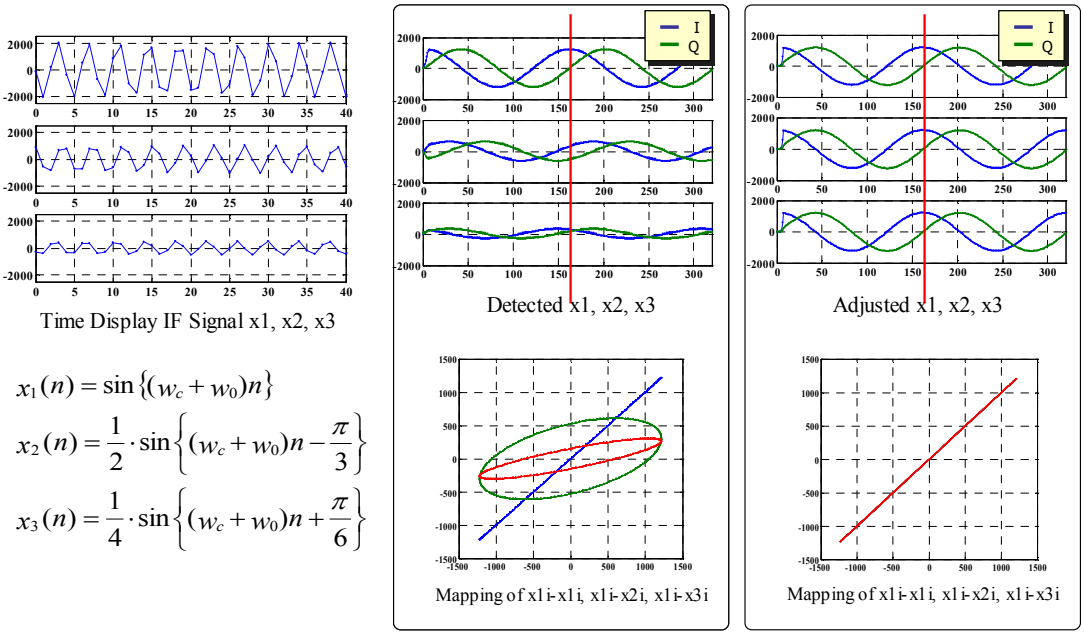


Fig. 3-12: Block Diagram of Adjustment by using MRC weight



$$x_1(n) = \sin\{(w_c + w_0)n\}$$

$$x_2(n) = \frac{1}{2} \cdot \sin\left\{(w_c + w_0)n - \frac{\pi}{3}\right\}$$

$$x_3(n) = \frac{1}{4} \cdot \sin\left\{(w_c + w_0)n + \frac{\pi}{6}\right\}$$

Fig. 3-13: Computer Simulation Result of Adjustment by NCO Control

3.5.3 Discussion

In this section, two digital calibration techniques were examined. The adjustment by NCO control is performed at IF stage. In this technique, an additional vector rotation circuit is not needed but the control values of NCO to adjust must be already known by any other computation process. In addition adjustment accuracy is dependent on a resolution of NCO sine/cosine table, that is, a large size of LUT is taken for an exact adjustment. On the other hand, the adjustment using the MRC optimum weight is performed at baseband. And additional vector rotation circuit is required but phase adjustment accuracy is relatively higher because it is indifferent to resolution of any table values stored in memory. The adjustment using MRC optimum weight is suitable for an FPGA implementation thanks to the simplicity.

3.6 Summary

In this chapter, the implementation of a simple digital phased array antenna using digital beamforming network was examined especially by using FPGAs as a DSP processor. It can steer a beam toward DOA of an incident signal automatically. This beamsteering function will provide more effective communication performance than fixed beam antenna. The circuit in FPGAs was implemented in simple parallel structure and the performance is expected to meet the requirement of realtime processing in practical wireless communications. The circuit has reasonable scale and can be expected to implement at low price. By some experiment in a radio anechoic chamber, the functionalities such as beamforming and DOA tracking are validated. This system can operate well in the range between -60° and +60° without any compensation or calibration. Finally, the digital approach to calibration of antenna response was introduced. For the present, it uses only for initialising the antenna response at any specific DOA.

Chapter 4

Examination of Applications Implementation: DOA Estimation By MUSIC Method

4.1 Introduction

Chapter 3 presented a beamforming application by an MRC beamforming antenna. That can steer and track automatically toward only one desired signal by using correlation between signals incoming at each element. Suppose that an adaptive antenna is only employed at the base station and not the mobile station, the mobile stations transmit using omni-directional antennas and therefore base-stations must employ spatially selective reception in order to separate the desired signals from interferences. As a matter of fact, when receiving communication signals at an adaptive antenna array, it is indispensable to estimate DOAs of incident signals and uses the DOAs in a beamformer in order to separate the desired signal. In this chapter the implementation issues in MUSIC (MUltiple SIgnal Classification) method, which is one of super resolution DOA estimation methods, and an examination of hardware design to implement on FPGAs are mentioned.

MUSIC is a subspace-based method. All the subspace-based methods are based on the EVD of the covariance (or correlation) matrix and the EVD is very difficult to process in realtime because of its heavy computational load and complexity. Section 4.5 describes the design of an EVD processor suitable for realtime processing hardware implementation. The architecture uses Cyclic Jacobi method based on CORDIC (COordinate Rotation DIgital Computer).

4.2 Performance Requirement

As mentioned before, in the communication environment multi-path fading caused by a reflection by any physical structures is a very serious problem. When passing through multi-path, the signals are delayed and out of phase from the signals through direct-path that causes the signal strength to be changed extremely at a receiver end and hence receiving quality is also reduced as shown in Fig. 4-1. The requirement of wider band and higher transmission rate in the next generation communications makes it more critical, furthermore when the mobile terminal moves at high speed the transmission rate may be confined by this problem. If can solve the multi-path fading, the trade off between mobility and transmission rate will be dramatically improved.

Adaptive antenna technologies can provide a solution of multi-path fading. The adaptive antenna can suppress the adverse effect of multi-path delayed coherent signals and interferences by steering beams toward intended direction and nulls the other undesired directions so that it can achieve high communication quality. This operation makes the receiving signal strength flat and stable over a threshold level. Therefore an adaptive antenna must find the DOAs of signals and form beams and nulls within a fading pitch. Considering mobility of several hundreds of *km*/*s*, the fading pitch becomes very short time, several μs. It is very difficult to compute them by a general serial structure DSP processor, and hence the high-speed parallel computations with a specific processor must be needed.

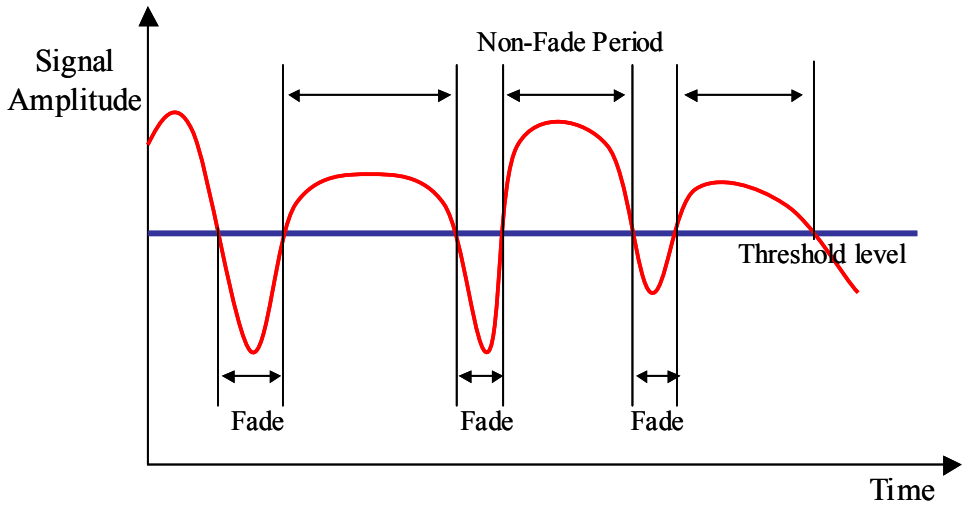


Fig. 4-1: Change of Received Signal Amplitude Under Multi Path Fading Environment

4.3 DOA Estimation using MUSIC Method

MUSIC (MUltiple SIgnal Classification) method is a kind of DOA (Direction Of Arrival) estimation algorithm based on eigenvector decomposition, which is also called based on spatial structure method. It has the capability to estimate DOA in much higher resolution than other conventional methods.

Spatial structure methods like MUSIC exploit the information in the steering vector $\boldsymbol{a}(\theta)$. The model of a signal $\mathbf{X}(t)$ received at K array elements are linear combinations of the L incident signals and noise. In this case, the steering vector is written by

$$\mathbf{a}(\theta_i) = \left[\exp(-j\frac{2\pi}{\lambda} d_1 \sin\theta_i) \quad \exp(-j\frac{2\pi}{\lambda} d_2 \sin\theta_i) \quad \dots \quad \exp(-j\frac{2\pi}{\lambda} d_K \sin\theta_i) \right]^T, \tag{4.1}$$

and matrix expression of $\mathbf{X}(t)$ is as following.

$$\begin{aligned} \mathbf{X}(t) &= \sum_{l=1}^{L} F_l(t) \cdot \mathbf{a}(\theta) + \mathbf{N}(t) \\ &= \mathbf{A}(\theta)\mathbf{F}(t) + \mathbf{N}(t) \end{aligned} \tag{4.2}$$

where $\mathbf{N}(t)$ is a noise vector and $\mathbf{A}(\theta)$ and $\mathbf{F}(t)$ are defined as (4.3) respectively.

$$\begin{aligned} \mathbf{A}(\theta) &= \left[\mathbf{a}(\theta_1) \quad \mathbf{a}(\theta_2) \quad \dots \quad \mathbf{a}(\theta_L) \right] \\ \mathbf{F}(t) &= \left[F_1(t) \quad F_2(t) \quad \dots \quad F_L(t) \right]^T \end{aligned} \tag{4.3}$$

The correlation matrix of $\mathbf{X}(t)$ is obtained by

$$\mathbf{R}_{XX} = E\left[\mathbf{X}(t)\, \mathbf{X}^H(t) \right] \tag{4.4}$$

(4.4) can be rewritten by substituting (4.2)

$$\mathbf{R}_{XX} = \mathbf{A} \cdot E\left[\mathbf{S}(t)\mathbf{S}^H(t) \right] \cdot \mathbf{A}^H + E\left[\mathbf{N}(t)\mathbf{N}^H(t) \right] \tag{4.5}$$

The correlation matrix of $\mathbf{S}(t)$ is denoted by $\mathbf{R}_{SS}$ and assuming $\mathbf{N}(t)$ is a White Gausian noise, the correlation matrix of $\mathbf{N}(t)$ is $\sigma^2\mathbf{I}$. Therefore $\mathbf{R}_{XX}$ can be written as

$$\mathbf{R}_{XX} = \mathbf{A} \cdot \mathbf{R}_{SS} \cdot \mathbf{A}^H + \sigma^2 \mathbf{I} \tag{4.6}$$

Since $\mathbf{R}_{XX}$ is a positive definite hermitian matrix, the correlation matrix of $x(t)$ can be decomposed by EVD(Eigen Value Decomposition) as

$$\mathbf{R}_{XX} = \mathbf{U} \cdot \mathbf{\Lambda} \cdot \mathbf{U}^H . \tag{4.7}$$

where $\mathbf{U}$ is an unitary matrix composed of eigenvectors and $\mathbf{\Lambda}$ is $diag\{\lambda_1, \lambda_2, ... \lambda_K\}$ of real eigenvalues ordered by $\lambda_1 \geq \lambda_2 \geq ..... \geq \lambda_K > 0$. If a vector $\mathbf{e}$ is orthogonal to $\mathbf{A}^H$ of rank L, that is, $\mathbf{e}$ is orthogonal to the range of $\mathbf{A}$ from (4.8), then $\mathbf{e}$ is an eigenvector of $\mathbf{R}_{XX}$ with eigenvalue of σ^2 as (4.9).

$$\mathcal{N}[\mathbf{A}^H] = {}^{\perp}\mathcal{R}[\mathbf{A}] \tag{4.8}$$

$$\mathbf{R}_{XX} \cdot \mathbf{e} = \mathbf{A} \cdot \mathbf{R}_{SS} \cdot \mathbf{A}^H \cdot \mathbf{e} + \sigma^2 \mathbf{e} = \sigma^2 \mathbf{e} \tag{4.9}$$

The eigenvectors of $\mathbf{R}_{XX}$ with eigenvalue of σ^2 lies in the nullspace of $\mathbf{A}^H$. On the other hand, there are some eigenvectors lying in the range of $\mathbf{A}$ and all eigenvalues are ordered as (4.10) partitioned into signal components and noise components.

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_L \geq \lambda_{L+1} = \ldots = \lambda_K = \sigma^2 \tag{4.10}$$

In the same manner the correlation matrix can be also partitioned as

$$\mathbf{R}_{XX} = \mathbf{U}_s \cdot \mathbf{\Lambda}_s \cdot \mathbf{U}_s^H + \mathbf{U}_n \cdot \mathbf{\Lambda}_n \cdot \mathbf{U}_n^H \tag{4.11}$$

where $\mathbf{U}_s$ and $\mathbf{U}_n$ are unitary matrices of signal subspace and noise subspace respectively and $\mathbf{\Lambda}_s$ and $\mathbf{\Lambda}_n$ are diagonal matrices of the eigenvalues.

The eigenvectors of noise space are orthogonal to the signal space, that is, orthogonal to signal steering vectors. If the noise space eigenvectors are used as a weight in beamforming, nulls steer toward the directions of incident waves. Using this principle, MUSIC method decomposes the correlation matrix and computes the spectrum using noise space eigenvectors as

$$P_{MU} \equiv \frac{\mathbf{a}^H(\theta) \cdot \mathbf{a}(\theta)}{\mathbf{a}^H(\theta) \cdot \mathbf{E}_N \mathbf{E}_N^H \mathbf{a}(\theta)} \tag{4.12}$$

$\mathbf{E}_N \equiv [\mathbf{e}_{L+1} \quad \cdots \quad \mathbf{e}_K]$: Noise subspace eigenvectors.

In the resulting spectrum, the peaks point the DOAs of incident waves because they are reciprocal of the nulls. Using null steering toward DOAs makes the super-resolution estimation possible [7]-[8].

4.4 Hardware Implementation of DOA Estimator using MUSIC Method

The computation flow of DOA estimation by MUSIC method is illustrated by Fig. 4-2. First, the correlation matrix is computed with data vector $\mathbf{X}(t)$, received at array antenna and time average of the correlation matrix is used to approximate a stochastic process. Then the spatial smoothing process suppresses the correlation between incident signals, which makes the estimation possible when the signals are correlated. The correlation matrix is decomposed into signal and noise sub-space and DOAs can be found by computing the angular spectrum with noise sub-space.

The computation of correlation matrix, spatial smoothing filter and spectrum synthesis by any fast algorithm can be implemented thanks to simplicity of their logics. But EVD problems are not so simple or rather complex. There are many algorithms for EVD problems but they are just numerical solutions for serial processing in general computer [9][10]. It is necessary to modify and reconstruct the serial algorithm suitable for realtime processing, the performance requirement of an adaptive antenna in the next generation communication system.

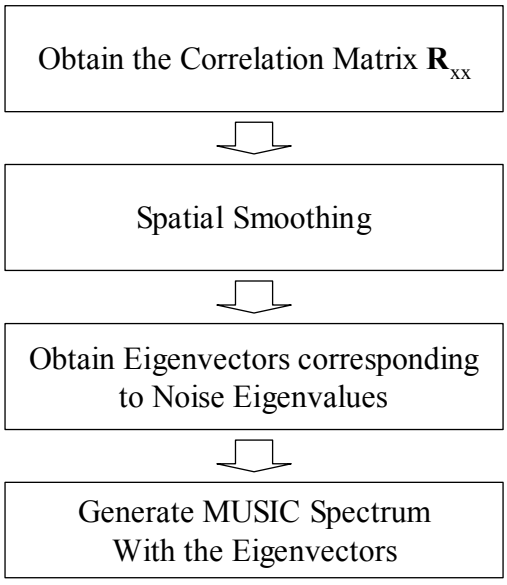


Fig. 4-2: Computational Flow of MUSIC Method

4.5 Design of EVD Processor using Jacobi Method Based On CORDIC Algorithm

Computing the Eigen Value Decomposition (EVD) of a symmetric matrix is a frequently encountered problem in adaptive array signal processing for super resolution DOA estimation method such as MUSIC, ESPRIT and etc. They are very difficult to implement due to the complexity and high performance requirement of EVD process, which needs huge scale of circuit to compute a number of MAC operations. So it is necessary to make these processes as simple as possible.

In this section an algorithm and the hardware friendly architecture for computing the EVD of a symmetric correlation matrix are proposed and the basic idea is also presented. Cyclic Jacobi method is well known for the simplest algorithm and easy implementation but its convergence time is slower than other factorization algorithm like QR-method [20]. But if considering the fast parallel computation of the EVD for hardware architecture with an ASIC or an FPGA, the Jacobi method is a good choice, since it offers a significantly higher degree of parallelism than QR-method and easy implementation [24].

In this section the EVD is computed using a Jacobi-type method, where the vector rotations are performed and the rotation angles are obtained by well-known CORDIC (COordinate Rotation DIgital Computer) algorithm. The hardware architecture suitable for ASIC or FPGA with fixed-point operation of only shift and add and the estimate of circuit scale are also introduced. Finally, the required precision for practical application to MUSIC DOA estimation is examined.

4.5.1 Cyclic Jacobi Method

Cyclic Jacobi method computes the symmetric eigenvalue problem by applying a sequence of orthonormal rotations to the left and right side of target matrix **A** as

$$\begin{gathered} \mathbf{V}^T \cdot \mathbf{A} \cdot \mathbf{V} = \mathbf{D}, \\ \left(\begin{array}{l} \because \mathbf{V} = \mathbf{J}_1 \cdot \mathbf{J}_2 \cdot \mathbf{J}_3 \cdots \\ \quad \mathbf{J}_k = \mathbf{P}_{12} \cdot \mathbf{P}_{13} \cdots \cdot \mathbf{P}_{n-1,n} \end{array} \right) \end{gathered} \tag{4.13}$$

where $\mathbf{P}_{pq}$ is an orthonormal plane rotation over an angle θ in the (p, q) plane and defined by (4.14) and **J** is multiple rotation of **P** in the cyclic-by-row manner and called a sweep of Jacobi rotation. Transforming matrix **A** infinitely, it is certain that **V** is the matrix whose column vectors are composed of eigenvectors and **D** is eigenvalue matrix whose diagonal elements are eigenvalues.

$$\mathbf{P}_{pq} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & \cos\theta & \cdots & \sin\theta & & \\ & & \vdots & 1 & \vdots & & \\ & & -\sin\theta & \cdots & \cos\theta & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{4.14}$$

A symmetric matrix **A** is transformed to **A**' by plane rotation as

$$\begin{aligned} \mathbf{A}' &= \mathbf{P}_{pq}^{T} \cdot \mathbf{A} \cdot \mathbf{P}_{pq} \\ &= \begin{pmatrix} & \cdots & a'_{1p} & \cdots & a'_{1q} & \cdots & \\ \vdots & & \vdots & & \vdots & & \vdots \\ a'_{p1} & \cdots & a'_{pp} & \cdots & a'_{pq} & \cdots & a'_{pn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a'_{q1} & \cdots & a'_{qp} & \cdots & a'_{qq} & \cdots & a'_{qn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ & \cdots & a'_{np} & \cdots & a'_{nq} & \cdots & \end{pmatrix}. \end{aligned} \tag{4.15}$$

The optimal rotation angle in a (*p, q*) plane is determined by

$$a'_{pq} = a'_{qp} = 0\,. \tag{4.16}$$

In above manner matrix **A** converges a diagonal eigenvalue matrix. It is called cyclic Jacobi method. In cyclic Jacobi method, defining the off-diagonal quantity $S^{(h)}$ by

$$S^{(h)} = \sqrt{\frac{1}{2}\left[\left\|\mathbf{A}^{(h)}\right\|_F^2 - \sum_{i=1}^{n}\left(a_{ii}^{(h)}\right)^2\right]}, \tag{4.17}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, the execution of a similarity transformation yields

$$S^{(h+1)} = \left[S^{(h)}\right]^2 - \left[\left(a_{pq}^{(h)}\right)^2 - \left(a_{pq}^{(h+1)}\right)^2\right], \tag{4.18}$$

Obviously the maximal reduction of $S^{(h)}$ is obtained if $a_{pq}^{(h+1)} = 0$. Therefore,

$$\lim_{h\to\infty} S^{(h)} \to 0 \quad \Leftrightarrow \quad \lim_{h\to\infty} \mathbf{A}^{(h)} \to diag[\lambda_1, \cdots, \lambda_n], \tag{4.19}$$

The condition for optimal angle, maximum reduction of $S^{(h)}$ is achieved as

$$\theta_{opt} = \frac{1}{2}\tan^{-1}\left[\frac{2a_{pq}}{a_{qq} - a_{pp}}\right] = \frac{1}{2}\tan^{-1}\tau, \tag{4.20}$$

where $\tau = \dfrac{2a_{pq}}{a_{qq} - a_{pp}}$.

4.5.2 CORDIC (COordinate Rotation DIgital Computer) Algorithm for Computing Vector Rotation

CORDIC (COordinate Rotation DIgital Computer) algorithm was published by Jack E. Volder in 1959 for the calculation of trigonometric functions, multiplication, division and conversion like rotation between binary radix number system [21]. The CORDIC algorithm provides an iterative method of performing vector rotations by arbitrary angles using only shifts and adds. CORDIC rotation can be explained from the basic vector rotation as

$$\begin{aligned}\mathbf{V}' = \begin{bmatrix} x' \\ y' \end{bmatrix} &= \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \\ &= \begin{bmatrix} x \cdot \cos\phi - y \cdot \sin\phi \\ x \cdot \sin\phi + y \cdot \cos\phi \end{bmatrix}\end{aligned}. \tag{4.21}$$

Fig. 4-3 illustrates the rotation of a vector $\mathbf{V} = [x\ y]^T$ by the angle ϕ.

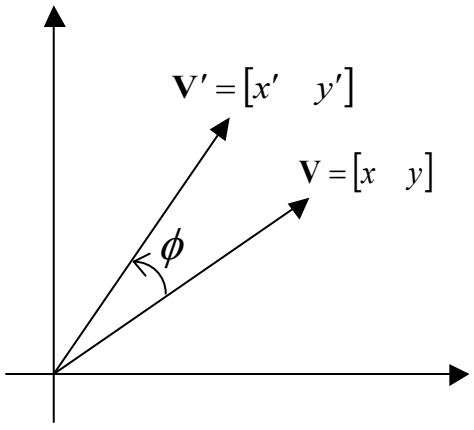


Fig. 4-3: Vector Rotation by the angle ϕ

(4.21) is rewritten as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \cos\phi \cdot \begin{bmatrix} x - y \cdot \tan\phi \\ y + x \cdot \tan\phi \end{bmatrix}. \tag{4.22}$$

The CORDIC procedure uses the basic angles ± $tan^{-1}(2^{-k})$ to compose the desired rotation angle ϕ. This algorithm is based on the decomposition of $\phi=z_0$ (z_0 is a initial value of z and here desired rotation angle) on the discrete basis $tan^{-1}(2^{-k})$ as written in (4.23).

$$\begin{aligned} &\tan\phi \equiv 2^{-i} \\ &\phi = \tan^{-1}2^{-i} \end{aligned} \quad : \quad \text{discrete angle} \tag{4.23}$$

Hence, after b+1 iteration steps the input vector $[x\ y]^T$ is rotated by ϕ with a precision of b bits. With $\phi = tan^{-1}(2^{-k})$ the $cos\phi$ can also be simplified as

$$\cos\phi = \frac{1}{\sqrt{1+\tan^2\phi}} = \frac{1}{\sqrt{1+2^{-2k}}} \tag{4.24}$$

and since even symmetry of cosine function it is a constant for a fixed number of iterations. The iterative expression can be obtained as

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = K_i \cdot \begin{bmatrix} x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_i + x_i \cdot d_i \cdot 2^{-i} \end{bmatrix}, \tag{4.25}$$

where $d_i = \pm 1$ (direction of rotation) and K_i is obtained by

$$K_i = \prod_{k=0}^{i} \frac{1}{\sqrt{1+2^{-2k}}} \quad : \quad \text{Scaling factor} \tag{4.26}$$

This scaling factor can be pre-calculated and applied elsewhere in the system. Normalizing the initial rotation vector by $|K|$, the gain of resulting vector inherent in the CORDIC algorithm is compensated. The resulting vector becomes the unit vector as shown in Fig. 4-4.

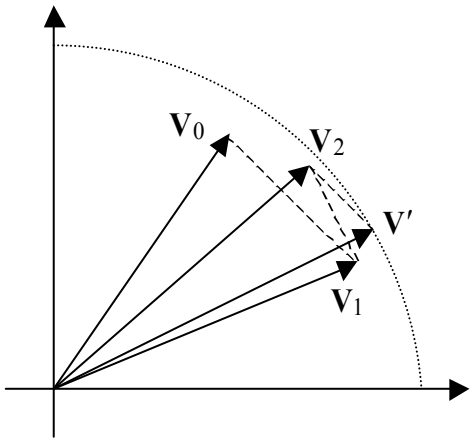


Fig. 4-4: Iterative Vector Rotation

The direction of each rotation is defined by d_i. It yields the third difference equation that determines the rotation direction and accumulates rotation angles as (4.27). Those angular values are supplied from a lookup table (LUT).

$$z_{i+1} = z_i - d_i \cdot \tan^{-1}(2^{-i}) \tag{4.27}$$

The CORDIC rotation is operated in one of two modes. The first mode, called rotation mode by Volder [21], rotates the input vector by a specified angle. The second mode, called vectoring mode, rotates the input vector to the x-axis while recording the angle required for that rotation. In rotation mode, the angle accumulator is initialized with the desired rotation angle. The rotation decision at each iteration step is made to decrease the magnitude of the residual angle in the angle accumulator and is therefore based on the sign of the residual angle after each step. For rotation mode, the CORDIC equations are

$$\begin{cases} x_{i+1} = x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_{i+1} = y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} = z_i - d_i \cdot \tan^{-1}(2^{-i}) \end{cases} . \tag{4.28}$$
$$d_i = -1 \text{ if } z_i < 0, +1 \text{ otherwise}$$

which provides the following result.

$$\begin{aligned} x_n &= A_n\left[x_0 \cos z_0 - y_0 \sin z_0\right] \\ y_n &= A_n\left[y_0 \cos z_0 + x_0 \sin z_0\right] \\ z_n &= 0 \\ A_n &= \prod_n \sqrt{1 + 2^{-2i}} \end{aligned} \tag{4.29}$$

In the vectoring mode, the CORDIC rotation rotates the input vector through whatever angle is necessary to align the result vector with the x-axis. The result of vectoring operation is a rotation angle and the scaled magnitude of the original vector in x component. The vectoring function works by seeking to minimize the y component of the residual vector at each rotation. For vectoring mode CORDIC equations are

$$\begin{cases} x_{i+1} = x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_{i+1} = y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} = z_i - d_i \cdot \tan^{-1}(2^{-i}) \end{cases} . \qquad (4.30)$$
$$d_i = +1 \text{ if } y_i < 0, -1 \text{ otherwise}$$

which provides the following result.

$$\begin{aligned} x_n &= A_n \sqrt{x_0^2 + y_0^2} \\ y_n &= 0 \\ z_n &= z_0 + \tan^{-1}(y_0 / x_0) \\ A_n &= \prod_n \sqrt{1 + 2^{-2i}} \end{aligned} \qquad (4.31)$$

The rotation and vectoring mode of CORDIC algorithm are limited to rotation angles between $-\pi/2$ and $\pi/2$ due to the use of 2^0 for the tangent in the first iteration [22]-[23].

4.5.2.1 Arctangent

The arctangent is directly computed using the vectoring mode CORDIC algorithm if the angle accumulator is initialized with zero (z_0=0). The argument must be provided as a ratio expressed as a vector $[x\ y]^T$. The result is taken from the angle accumulator as

$$z_n = z_0 + \tan^{-1}(y_0 / x_0) . \qquad (4.32)$$

4.5.2.2 Circuit Architecture

The CORDIC algorithm only performs shift and add operations and is therefore easy to implement and suitable for circuitry. There can be various architectures according to balancing circuit complexity with respect to performance. The most obvious methods of implementing a CORDIC algorithm are bit-serial, bit-parallel, unrolled, iterative, and etc. This paper uses the bit-parallel unrolled CORDIC architecture for high performance. It is a cascade structure of the

iterative CORDIC stages as shown in Fig. 4-5. The unrolled design only consists of combinatorial components and computed one result per a clock cycle without sequential logic.

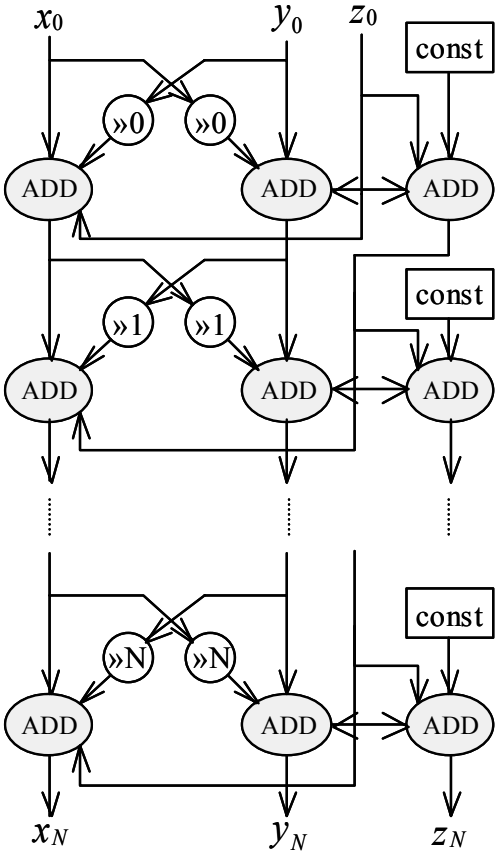


Fig. 4-5: Unrolled CORDIC Architectures

4.5.2.3 Scaling by Double Rotation

In CORDIC process, a scaling must be performed. As described in previous subsection the scaling method using pre-computed scaling value and normalization couldn't perform only with shift and add operations. The double rotation by $\phi/2$ can solve this problem.

Let an elementary rotation by angle ϕ be composed of twice executing a rotation by $\phi/2$. Hence (4.28) is rewritten as

$$\begin{cases} x_{k+1} = (1-2^{-2k})x_k - y_k \cdot d_k \cdot 2^{-k+1} \\ y_{k+1} = (1-2^{-2k})\, y_k + x_k \cdot d_k \cdot 2^{-k+1} \\ \quad z_{k+1} = z_k - d_k \cdot \tan^{-1}(2^{-i}) \end{cases} . \qquad (4.33)$$

$$d_k = -1 \text{ if } z_k < 0, +1 \text{ otherwise}$$

It requires 4 shifts and 6 adds operations per iteration stage as shown in Fig. 4-6 but the scaling factor becomes square root free. In addition, it gives the simple scaling factor to avoid the division operation because for a given precision b of the adders all factors $(1-2^{-k})$ with $k>b$ do not contribute to K_k. So the scaling factor is simplified to (4.34) [24].

$$K_k = \prod_{i=0}^{k} \frac{1}{1+2^{-2i}} \approx \frac{1}{2} \prod_{i=1}^{k/4} (1-2^{-(4i-2)}) \tag{4.34}$$

The scaling can be also computed only with shift and add operation. Fig. 4-6 shows the k-th stage of the double rotations. In this paper twice executing rotation architecture was introduced for circuit efficiency.

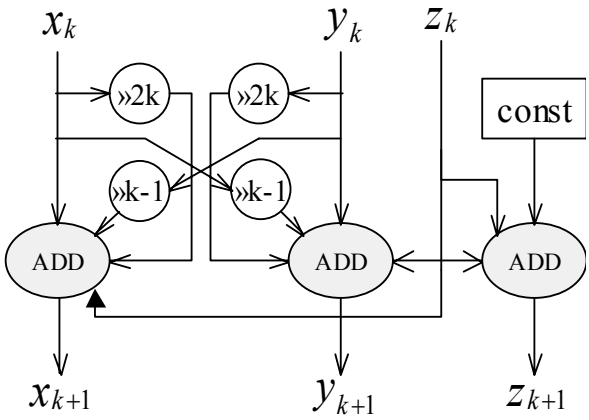


Fig. 4-6: Double rotation k-th stage

4.5.3 Examination of CORDIC-Jacobi EVD Processor with Fixed Point Operation

This subsection describes the practical implementation of Jacobi EVD processor based on CORDIC with a fixed-point operation including required number of Jacobi sweeps, precision for desired accuracy and application to MUSIC DOA estimator.

The precision of fixed-point operation is the first thing to determine. Fig. 4-7 illustrates the reduction of off-diagonal norm as iteration number of Jacobi sweeps with random real symmetric matrix of 12-bit precision as an input. Generally in Jacobi method 6~10 sweeps are sufficient to achieve convergence with floating-point operation [20]. But Fig. 4-7 confirms that the 32-bit floating-point operation converges to the machine zero within a given precision by 4~5 Jacobi sweeps, but fixed-point operations not converge but only keep on vibrating after 4~5 Jacobi sweeps regardless of the precision from 12 to 36 bits. This is caused by the limit-precision of the fixed-point operation It is the trade-off between implementation simplicity, circuit scale, performance and accuracy. Without using additional decision circuit, determining the number of sweeps to be 4~5 fixed is a proper choice and more computations are needless for an efficiency in hardware architecture.

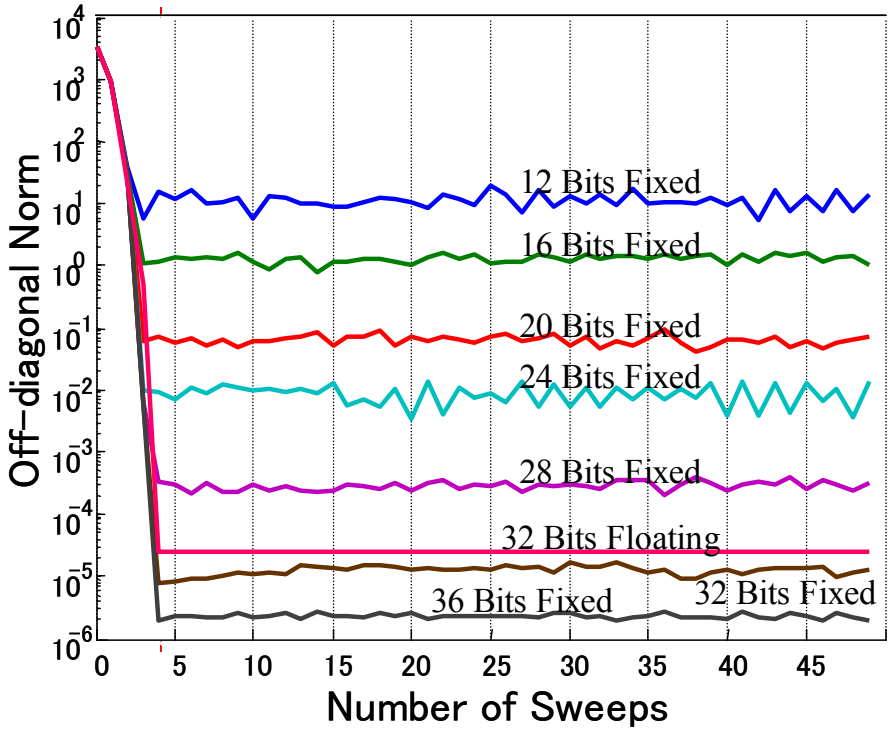


Fig. 4-7: Reduction of Off-diagonal Norm

The next thing to examine is the precision of fixed-point operation. To validate fixed-point operation, the accuracy within allowable error range must be guaranteed. (4.35) yields the error ratio where **v**'s are the vectors whose elements consist of eigenvalues computed in respective subscript ways and $\|\cdot\|$ denotes vector norm. Physically the error between vectors is defined as Fig. 4-8.

$$\text{Error} = \frac{\left\|\mathbf{v}_{\text{float}} - \mathbf{v}_{\text{fixed}}\right\|}{\left\|\mathbf{v}_{\text{float}}\right\|} \times 100(\%) \tag{4.35}$$

Fig. 4-9 shows the error ratio of fixed-point operations on the architecture mentioned in previous subsections to the 32-bit floating-point operation. Fig. 4-9 used a random real symmetric matrix of 12-bit precision as an input.

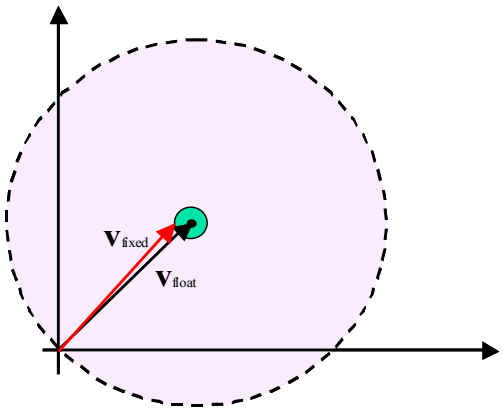


Fig. 4-8: Errors between Vectors

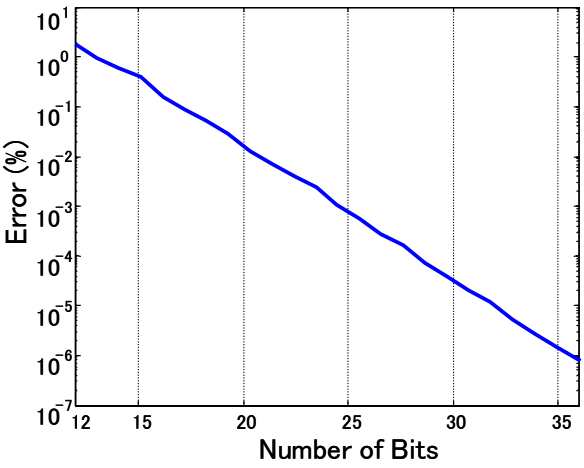
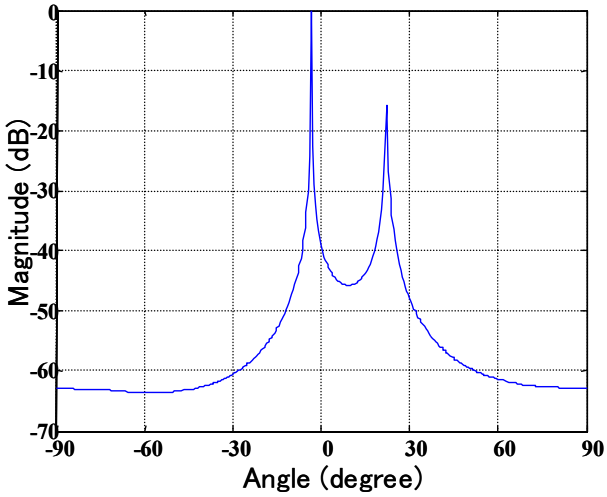
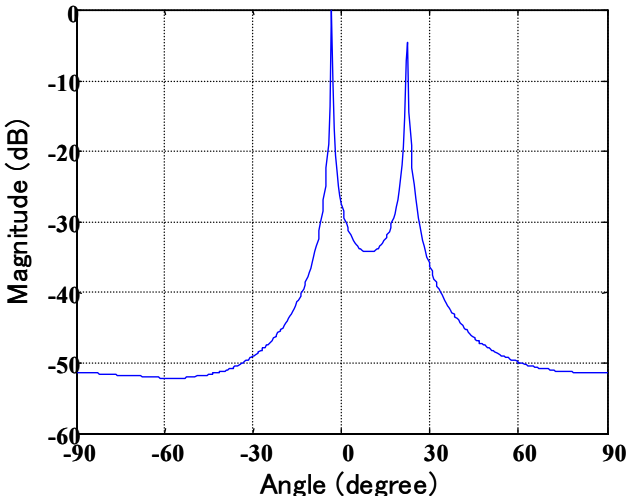


Fig. 4-9: Error Ratio of fixed-point operations to 32 bits floating point operation

For practical uses, less or equal to 16-bits precision will be desired. When it is implemented with 16-bits precision, the error ratio becomes about 0.14%. Fig. 4-10 is the results of DOA estimation by spectral MUSIC method supposing 4 elements array antenna and 2 incident waves. The EVD computations are performed with 32-bit floating-point operation as Fig. 4-10 (a) and 16-bit fixed-point operation using Jacobi EVD method based on CORDIC which is the proposed architecture as Fig. 4-10 (b) respectively. It is difficult to find the estimation accuracy because of using spectral MUSIC but the results show that the fixed-point operation of 16-bit precision performs well estimation by MUSIC spectrum. More exact allowable error range to the precision of the fixed-point operation may be found by using numerical algorithm like Root-MUSIC.



(a)



(b)

Fig. 4-10: MUSIC Spectrum in case of 4 elements and 2 incident waves
(DOAs are -5° and 20°)
(a) 32-bit floating-point operation and (b) 16-bit fixed-point operation by CORDIC-Jacobi EVD

4.5.4 Hardware Design of EVD Processor

In previous subsections it was mentioned that 4~5 Jacobi sweeps are sufficient for convergence and the fixed-point operation of 16-bit precision gives a good enough spectral estimation result. Based on these facts, this subsection proposes hardware circuit architecture of the EVD processor and the design considering the implementation on FPGAs. Fig. 4-11 show the computational flow of this method. As described before, Jacobi type EVD process is very simple and just a sequence of vector rotations. From (4.20) the optimal rotation angles are determined and then the processor performs the similarly conversion of correlation matrix **R** and unitary matrix **E** of eigenvectors. After 4~5 sweeps, the computation is complete. The resulting matrix **R** converges the diagonal matrix of eigenvalues and **E** becomes the unitary matrix of eigenvectors.

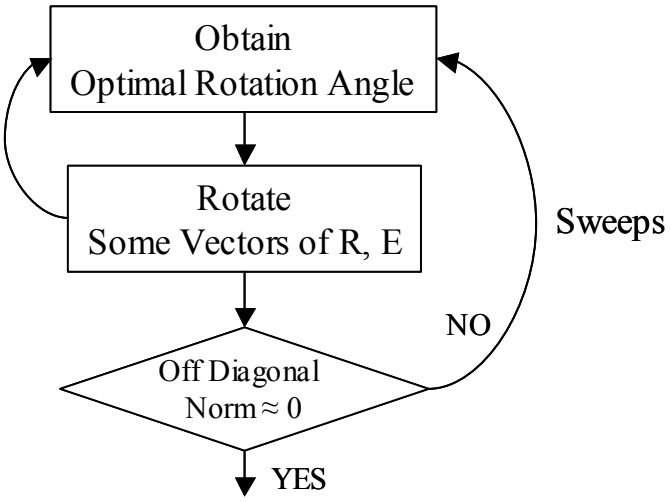


Fig. 4-11: Computation Flow

The EVD processor consists of CORDIC matrix rotators and CORDIC arctangent. With the optimum angles obtained by (4.20), the rotation $\mathbf{P}_{pq}$ is determined as (4.14). The transform $\mathbf{P}_{pq}$ in (4.15) changes only p, q rows and columns of the matrix **A**. Therefore the transform can be simplified as

$$\begin{aligned}\begin{pmatrix}\mathbf{a}'_p\\ \mathbf{a}'_q\end{pmatrix} &= \mathbf{p}_{pq}^{T}\cdot\begin{pmatrix}\mathbf{a}_p\\ \mathbf{a}_q\end{pmatrix}\\ &= \begin{pmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{pmatrix}\cdot\begin{pmatrix}a_{p1} & \cdots & a_{pp} & \cdots & a_{pq} & \cdots & a_{pN}\\ a_{q1} & \cdots & a_{qp} & \cdots & a_{qq} & \cdots & a_{qN}\end{pmatrix},\end{aligned} \tag{4.36}$$

where $\mathbf{a}_k$ is the k-th row vector of the matrix **A** and $\mathbf{p}_{pq}$ is the vector rotation, a submatrix of $\mathbf{P}_{pq}$ in (4.14). Thanks to the symmetry of the matrix **A**, the right side transform yields the same result so it is not necessary any more except the second rotation of (a_{pp}, a_{qq}) of **A**. The diagram of EVD processor is illustrated in Fig. 4-12, where the ESB (Embedded System Block) is a memory block of an FPGA and stores the correlation matrix and eigenvector matrix. In the CORDIC matrix rotator, N CORDIC vector rotators are arranged in parallel. It takes $N(N\text{-}1)/2$ matrix rotations per one sweep to computing **R** and **E** respectively. Fig. 4-13 illustrates the architecture of the EVD processor core.

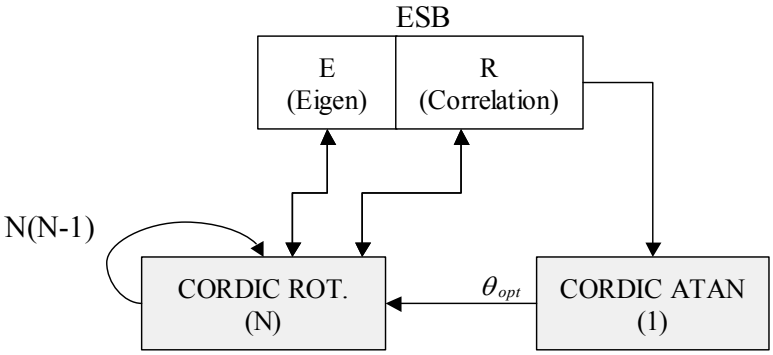


Fig. 4-12: Block Diagram of EVD Processor

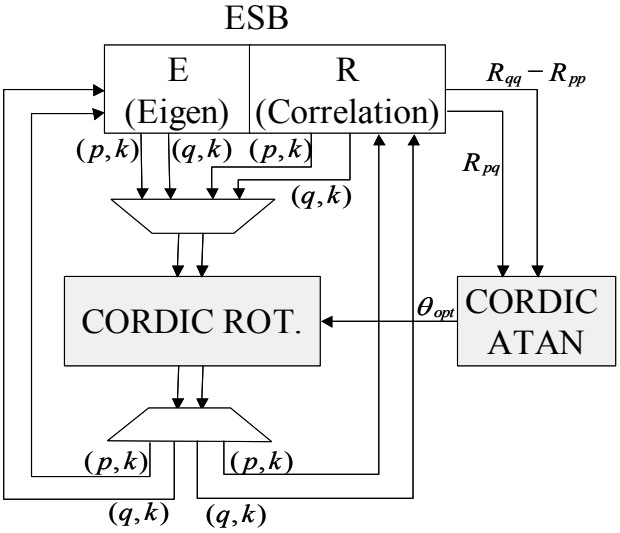


Fig. 4-13: Architecture of EVD Processor Core

4.5.5 Circuit Scale / Expected Performance

Basic operation of the proposed EVD processor consists of only shifts and adds. The computational load required for a CORDIC stage with the double rotation is $4B + 1/4B$ (needed for scaling operation) shift and $6B + 1/4B$ (needed for scaling operation) add where B is bit length. The number of vector rotations required computing both eigenvalues and eigenvectors is $4\times N(N\text{-}1)(N\text{+}2)$ where 4 Jacobi sweeps and N is the double number of array length. Hence total computational load is $[4\times N(N\text{-}1)(N\text{+}2)\text{+}1]$* (17/4$B$ Shift + 25/4B Add).

On the other hands, if configuring this computation load with the parallel architecture proposed in previous subsections, rough estimate of circuit scale is about $(N\text{+}1)\times$15K equivalent gates. For example, if N=8 (4-element array antenna if $n\times n$ correlation matrix of complex numbers is converted into $2n\times 2n$ matrix of only real numbers [20]) the total circuit scale is about 75K equivalent gates.

This system, if arranged parallel structure, requires $[4\times N(N\text{-}1)\times 2\text{+}1]$*$(B\text{+}1)$ clock cycles to compute one EVD computation. For example, if N=8, B=16 and operated at the speed of 100 MHz, this system can compute EVD computations about 13,200 times (75.8 μs / EVD). The EVD is the most dominant process in the whole processing load from DOA finding to beamforming. This fast computation processor must provide efficient use.

4.5.6 Discussion

In this section the circuit design of EVD computation processor for MUSIC DOA estimator was proposed. It uses CORDIC based Jacobi method and is suitable for hardware implementation for realtime processing. Adopting fixed-point operation causes some error but make the implementation easy and achieve the high performance. In addition, the accuracy of the 16-bit fixed-point operation has the error less 0.14% and the functionality for spectral MUSIC method could be confirmed.

4.6 Summary

This chapter described the DOA estimation application of an adaptive antenna technology. The MUSIC method has super resolution of estimating DOAs of incident waves but is based on complex EVD computation. The principle of MUSIC method was presented and the performance requirement for multi-path fading solution was discussed. The hardware implementation on an FPGA of MUSIC DOA estimator consists of lots of processes, the most complex process of which is the EVD computation and it dominates the whole system throughput. This chapter proposed the Jacobi-type EVD processor based on CORDIC algorithm. It is suitable for hardware architecture of parallel processing and FPGAs provide the efficient configuration.

Chapter 5

Summary And Conclusion

This chapter summarizes and concludes the paper.

This paper introduced the digital prototype system for the evaluation of an adaptive antenna based on FPGAs considering the reconfigurablity of software defined radio (SDR) and the performance of realtime processing required to an adaptive antenna system.

As an application, the implementation of 2-element MRC (Maximal Ratio Combining) beamforming receiver similar to a phased array antenna in analog sense using of digital beamforming network was examined. The circuit in FPGAs was implemented in parallel structure and the performance is expected to meet the requirement of realtime processing in practical wireless communications. By some experiment in a radio anechoic chamber, the functions of such as beamforming and DOA tracking are validated. In addition, two digital calibration techniques were proposed. The adjustment method by NCO control, which is performed at IF stage and the other method using the MRC optimum weight which is performed at baseband were discussed. They are easy to implement on FPGAs and can be processed in realtime.

As the other application, this paper described the DOA estimation technique. The MUSIC method has super resolution estimating DOAs of incident waves but is based on complex EVD computation, which dominates the whole system load. The Jacobi-type EVD processor based on CORDIC algorithm was proposed, which is suitable for hardware architecture like parallel processing and FPGAs provide the efficient configuration. It is expected to meet the performance requirement for the next generation wireless communication.

The cost of hardware will become reasonable and the performance will be also improved as a progress of device technologies. The digital technology is replacing an old fashion of analog technology step by step. An adaptive antenna which combines antenna technology and digital signal processing technology will be quite promising in the next generation communication system.

Acknowledgments

I thank many people who supported me in their helpful comments and enlightening suggestions. First of all, my supervisor, associate professor Dr. Hiroyuki Arai always gave me the courage to perform my research abroad, in Japan as well as the technical supports. Dr. Masahiro Fukuta of Brains Corp. gave me the opportunity to participate in this research. He also helped me the hardware support (manufacturing prototype system) and guided the hardware implementation. And I also appreciate Dr. Koichi Ichige for a plenty of advices and criticism about my research. I would like to extend my gratitude to everybody in prof. Arai and prof. Ichige laboratory.

Finally, especially I thank to my parents and my family in Korea.

Bibliography

[1] R. Miura, T. Tanaka, A. Horie and Y. Karasawa, "A DBF Self-Beam Steering Array Antenna for Mobile Satellite Applications Using Beam-Space Maximal-Ratio Combination," IEEE trans on vehicular technology, Vol. 48, No 3, May 1999

[2] Per H. Lehne and Magne Pettersen, "An Overview of Smart Antenna Technology for Mobile Communications Systems," IEEE Communications Surveys, Fourth Quarter 1999, Vol. 2, No. 4, http://www.comsoc.org/pubs/surveys

[3] Y. Karasawa, "The Software Antenna: A New Concept of Kaleidoscopic Antenna in Multimedia Radio and Mobile Computing Era," IEICE Trans., Comm., Vol. E80-B, No. 8, pp.1214-1217, Aug. 1997

[4] H. Steyskal, "Digital Beamforming Antennas, An Introduction," Microwave Journal, Vol. 30, No. 1, pp. 107-124, Jan. 1987

[5] J. Mitola III, "Software Radios Survey, Critical Evaluation and Future Directions," IEEE National Telesystems Conference, pp. 13-15~13-23, 1992

[6] G. Prescott and S. Gurrapu "Design And Implementation of a DSP Digital Beam Forming Receiver," Information and Tele-communication Technology Center, University of Kansas, 1997 ICSPAT Proceeding

[7] N. Kikuma, Adaptive Signal Processing with Array Antenna, Science and Technology Publishing Company, Inc., 1999, in Japanese

[8] Jens Baltersee, "Development and Analysis of Protocols for Directed antennas in the Mobile Broadband System,"
http://www.comnets.rwth-aachen.de/~ftp-mbs/publications/dir_ant_da/da.html

[9] Simon Haykin, Adaptive filter theory, Third Edition, Prentice Hall, 1996

[10] Gene H. Golub and Charles F. Van Loan, Matrix Computations, The Johns Hopkins University Press, 1996

[11] B. Fawcett and R. Iwanczuk, "Techniques and Tools for FPGA-based DSP design," Xilinx Inc.

[12] Mark Cummings and Shinichiro Haruyama, "FPGA in the Software Radio," IEEE Communication Magazine, Vol. 37, No. 2, pp.108-112, Feb. 1999

[13] R. Baines, “The DSP Bottleneck,” IEEE Communications Magazine, May 1995, pp. 46~54

[14] John Litva, Titus Kwok-Yeuong Lo, Digital Beamforming in Wireless Communications,” Artech House Publishers, 1996

[15] M. Kim, H. Arai and M. Fukuta “Development of a Digital Prototype Board for Adaptive Array Antenna Receiver,” Submitted to 2001 IEICE Society Conference

[16] John G. Proakis, Digital Communications, 4th ed, McGraw-Hill, 2001

[17] Digital IF Subsampling Using the HI5702, HSP45116 and HSP43220, Intersil Application Note, AN9509.2

[18] Implementing FIR filters in FLEX Devices, ALTERA Applications Note 73, Feb. 1998

[19] Rodney G. Vaughan, Neil L. Scott, D. Rod White, “The theory of bandpass sampling,” IEEE Trans. on signal processing, Vol. 39, No. 9, pp. 1973~1984, Sep. 1991

[20] William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery, Numerical Recipes in C, The Art of Scientific Computing, Second Edition, Cambridge University Press, 1992

[21] Volder J. E. “The CORDIC trigonometric computing technique,” IRE Trans. Electronic Computing, Vol. EC-8, 1959

[22] Muller and Jean Michel, Elementary Functions: Algorithms and Implementation

[23] Andraka R. “A survey of CORDIC algorithms for FPGA based computers,” Proceeding of the 1998 ACM/SIGDA sixth international symposium on Field Programmable Gate Arrays, pp 192-200, Monterey, CA, Feb. 1998

[24] Jurgen Gotze, Steffen Paul and Matthias Sauer, “An Efficient Jacobi-like Algorithm for Parallel Eigenvalue Computation”, IEEE Trans. On Computer, Vol. 42, No. 9. Sep. 1993

Publication List

IEICE Technical Report

- “FPGA-Based DSP Implementation of Simple MRC Digital Beamforming Antenna"
  Minseok Kim, Koichi ICHIGE, Hiroyuki ARAI,
  IEICE Antenna &Propagation Technical Report, AP2001-49,
  Kitami Institute of Technology, Hokkaido, JAPAN, Jul. 2001

IEICE General Conference

- “Examination of FPGA Implementation on Eigenvalue Decomposition in MUSIC Method”,
  Minseok Kim, Koichi ICHIGE, Hiroyuki ARAI,
  2001 IEICE General Conference, B-1-45,
  Ritsumeikan Univ., Kyoto, JAPAN, Mar. 2001

- “Development of a Digital Prototype Board for Adaptive Array Antenna Receiver”
  Minseok Kim, Hiroyuki ARAI, Masahiro FUKUTA,
  2001 IEICE Communication Society Conference, B-1-151,
  The University of Electro-Communications, Chofu, JAPAN, Sep. 2001

- “Design of EVD Computation Processor for MUSIC DOA Estimation By using Cyclic Jacobi Method based on CORDIC”
  Minseok Kim, Koichi ICHIGE, Hiroyuki ARAI,
  2002 IEICE General Conference, A-4-70 (Accepted)
  Waseda University, Tokyo, Japan, Mar. 2002

International Conference

- “FPGA-Based DSP Implementation of Simple MRC beamformer”
  Minseok Kim, Koichi ICHIGE, Hiroyuki ARAI
  APMC 2001 Conference, pp.589-592,
  National Taiwan University, Taipei, TAIWAN, Dec. 2001