EFFICIENT TECHNIQUES FOR HARDWARE IMPLEMENTATION OF CONSTANT MODULUS ALGORITHM (CMA) ADAPTIVE ANTENNA

AKKARAT BOONPOONGA

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR THE DEGREE OF
DOCTOR OF ENGINEERING IN ELECTRICAL ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2008

KMITL-2008-EN-D-018-036

หัวข้อวิทยานิพนธ์	เทคนิคที่มีประสิทธิภาพสำหรับการสร้างฮาร์ดแวร์ของสายอากาศแถวลำดับปรับตัวที่ใช้อัลกอริทึม CMA
นักศึกษา	นายเอกรัฐ บุญภูงา
รหัสประจำตัว	47060003
ปริญญา	วิศวกรรมศาสตรดุษฎีบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2551
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ศ.ดร.โมไนย ไกรฤกษ์
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	ผศ. ดร. เผ่าภัค ศิริสุข

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอเทคนิคสองเทคนิคที่ใช้เพิ่มประสิทธิภาพของสายอากาศปรับตัวที่ใช้อัลกอริทึม CMA โดยเทคนิคแรกคือการเพิ่มอัตราการลู่เข้าของ CMA เทคนิคนี้จะใช้ตัวเลื่อนเฟสหนึ่งบิตและตัววัดกำลังงานซึ่งปรกติจะถูกรวมไว้ในสายอากาศหลายลำคลื่น เพื่อหาทิศทางของลำคลื่นที่มีกำลังงานสูงสุด ลำคลื่นดังกล่าวจะถูกใช้เป็นลำคลื่นเริ่มต้นสำหรับกระบวนการปรับตัว CMA ระบบการเริ่มต้นโดยใช้ฮาร์ดแวร์ที่นำเสนอนี้จะถูกพัฒนา อีกทั้งยังสร้างส่วนประมวลผลปรับตัว CMA และวงจรควบคุมที่เกี่ยวข้องโดยใช้ FPGA ประสิทธิภาพของระบบการเริ่มต้นที่นำเสนอจะถูกประเมินโดยการจำลองระบบ ตัวต้นแบบระบบสายอากาศ ที่นำเสนอจะถูกทดสอบร่วมกับสายอากาศแถวลำดับปรับเฟสที่แบนกะทัดรัดสี่ลำคลื่นที่ถูกออกแบบสำหรับทำงานในย่านความถี่ 1.95 GHz ของระบบ IMT2000 ผลการทดลองจะยืนยันความสามารถของเทคนิคที่นำเสนอ นอกจากนี้ สายอากาศแถวลำดับปรับเฟสลำคลื่นขององค์ประกอบปรับได้จะถูกใช้แทนสายอากาศแถวลำดับปรับเฟสที่แบนกะทัดรัดสี่ลำคลื่น สายอากาศนี้สามารถสร้างลำคลื่นที่สามารถใช้เริ่มต้นได้ 12 ลำคลื่นโดยการใช้ตัวปรับเฟสหนึ่งบิตและการประยุกต์ใช้การไบแอสไปหน้าและย้อนต่อไดโอดพินของแต่ละองค์ประกอบ ผลการจำลองจะถูกแสดงเพื่อยืนยันว่าระบบสายอากาศปรับตัวที่นำเสนอร่วมกับสายอากาศแถวลำดับปรับเฟสลำคลื่นขององค์ประกอบปรับได้มีอัตราการลู่เข้าเร็วที่สุดเมื่อเปรียบเทียบกับระบบที่ใช้สายอากาศแถวลำดับปรับเฟสที่แบนกะทัดรัดสี่ลำคลื่นและไม่มีการเริ่มต้นโดยการปรับลำคลื่น เทคนิคที่สองคือการเพิ่มประสิทธิภาพของหน่วยประมวลผลปรับตัว CMA โดยนำเสนอสถาปัตยกรรมการคำนวณแบบขนาน (parallel) และท่อ (pipelining) อัลกอริทึม CMA จะถูกปรับให้เป็นอัลกอริทึม DCMA โดยการแทรกตัวหน่วง (delay unit) เข้าไปในอัลกอริทึม CMA จะศึกษาผลกระทบของการแทรกตัวหน่วงและความยาวของคำ (wordlength) ต่อสายอากาศแถวลำดับปรับตัวที่นำเสนอผ่านการจำลอง สถาปัตยกรรมที่นำเสนอ

ของอัลกอริทึม DCMA จะถูกสร้างโดยใช้การดำเนินการแบบจุดตายตัว (fixed-point arithmetic) บน FPGA นอกจากนี้วงจรคูณแบบจุดตายตัวจะถูกแทนที่ด้วยวงจรคูณแบบฐานยกกำลังเป็นสอง (power-of-two multipliers) เพื่อลดการใช้ทรัพยากรในการสร้าง FPGA ผลการทดสอบจะแสดงให้เห็นว่าหน่วยประมวลผล CMA ที่ถูกสร้างสามารถทำงานร่วมกับระบบสายอากาศแถวลำดับปรับตัวที่นำเสนอ

Thesis	Efficient Techniques for Hardware Implementation of Constant Modulus Algorithm (CMA) Adaptive Antenna
Student	Mr. Akkarat Boonpoonga
Student ID.	47060003
Degree	Doctor of Engineering
Program	Electrical Engineering
Year	2008
Thesis Advisor	Prof. Dr. Monai Krairiksh
Thesis Co-advisor	Assist. Prof. Dr. Phaophak Sirisuk

ABSTRACT

In this thesis, two techniques which can enhance the performance of an adaptive array antenna using a constant modulus algorithm (CMA) are proposed. Firstly, a cost-effective approach for initialization of the CMA adaptive antenna is presented to increase the convergence rate of CMA. The technique utilizes one-bit phase shifters and a power detector, which are normally integrated with a multi-beam antenna, to determine the maximum power beam direction. Consequently, the beam is exploited as an initial beam for CMA. Development of hardware-assisted initialization is discussed. In addition, FPGA implementation of a CMA processor and associated control circuitry is presented. Several simulations are performed to evaluate the performance of the proposed initialization technique. In addition, the developed prototype is tested with a flat four-beam phased array antenna designed for the operation at the IMT2000 frequency of 1.95 GHz. Experimental results confirm superiority of the proposed technique. Besides, a new antenna, a switched-beam element phased array antenna, is employed instead of a flat four-beam phased array antenna. It can generate twelve possible initial beams, covering the service area (360°) by using four one-bit phase shifters and applying forward or reverse bias to PIN diodes of each element. The simulation results are shown to validate that the proposed adaptive antenna system with the switched-beam element phased array antenna has the fastest convergence rate compared to that obtained from other systems, namely the systems with flat four-beam phased array antenna and without beam-switching initialization. Secondly, a technique which can improve the speed performance of a CMA adaptive processing unit is presented by using parallel and pipelining processing. The original CMA is modified to be a delayed CMA (DCMA) by inserting

delay units. The effect of inserting delay units and wordlenght on the proposed adaptive antenna system is investigated via simulations. The proposed pipelined architecture based on DCMA is implemented by using fixed-point arithmetic on a field programmable gate array (FPGA) and then tested. In addition, fixed-point multipliers are replaced by power-of-two multipliers to reduce the use of resource for FPGA implementation. The testing results show that the proposed adaptive processing unit can operate well with the proposed adaptive antenna.

ACKNOWLEDGEMENTS

This thesis is the result of doing research hard whereby I have been accompanied and supported by many people. It is a pleasant aspect that I have now the opportunity to express my gratitude for all of them.

First, I would like to express my sincere gratitude to my advisor, Professor Monai Krairiksh, for his invaluable ideas, and supervision. I am deeply grateful to my co-advisor, Assistant Professor Phaophak Sirisuk, head of Department of Computer Engineering, Mahanakorn University of Technology, Bangkok, Thailand, for his suggestions and guidance in my work. Their understanding and encouraging guidance have provided a good basis and contributed to the success and completion of this thesis.

My thanks also go to Associated Professor Sompol Kosulvit and Assistant Professor Chuwong Phongcharoenpanich for their helpful discussions.

I would like to acknowledge Mr. Watchara Pakkamart at Department of Electrical Engineering, King Mongkut's Institute of Technology NorthBangkok, Bangkok, Thailand, for his kindness to lend an FPGA evaluation kit for a while.

I wish to thank all people in Wireless Communication Laboratory for their kind supports and providing a friendly atmosphere to work.

I would like to thank Professor Tapan K. Sarkar at Department of Electrical Engineering and Computer Science, Syracuse University, USA for giving an invaluable opportunity to be a visiting scholar in his laboratory. Besides, I wish to thank all people in that laboratory for a friendly atmosphere and teaching English to me. Special thanks go to Mr. Santana Buritramart for his great assistance when I was in Syracuse.

I thank the Thailand Research Fund (TRF) which provides financial assistance to the undertaken project through the Royal Golden Jubilee Ph.D. Program (Grant No. PHD/0011/2548). Also, the partial support offered by the National Electronics and Computer Technology Center (NECTEC), the National Science and Technology Development Agency (NSTDA), Thailand, under Project Number: NT-B-22-T2-38-47-13 and Research No.: 13/2547 and by the Thailand Research Fund under contract No. RTA4880002 is gratefully acknowledged.

Finally, I wish to express my gratitude to my family for their love, understanding and encouragement throughout my life.

Akkarat Boonpoonga

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

CHAPTER 1
INTRODUCTION

1.1 Background

One of the fastest growing fields in the engineering world is the wireless communication system because of the rapid increase in the demand for its services. It was anticipated that people will be able to use a mobile device anywhere on the globe at all time in the near future. The way to achieve the anticipation is to increase capacity by mounting antennas on vehicles, ships, aircraft, satellites and basestations. This is expected to play an important role in fulfilling these services. However, the limitation to increase capacity was encountered.

The higher demand in wireless communications calls for higher system capacities. The capacity of a communication system can be increased directly by enlarging the bandwidth of the existing communications channels or by allocating new frequencies to the service. However, the electromagnetic spectrum which is a valuable resource is limited and the electromagnetic environment is increasingly congested with a proliferation of unintentional and intentional sources of interference. Thus, the direct method to increase the system capacity by opening the new spectrum space is not feasible in the future. The use of the frequency resource will be critical. Other methods to improve the system capacity were introduced by using multiple access techniques. The techniques refer to the simultaneous transmission by numerous users through a common receiving point which is normally the basestation. There are four domains where the users can share the use of the basestation, namely bandwidth, time, code and space. The examples of the multiple access techniques include frequency-division multiple access (FDMA), time-division multiple access (TDMA), code-division multiple access (CDMA), and space-division multiple access (SDMA) techniques. In the FDMA scheme, the frequency spectrum is divided into segments that are apportioned among difference users. With the use of digital techniques, TDMA became a practical access technology. Here, each user is apportioned the entire transmission resource periodically for a short period of time. Digital techniques enable another multiple access method which is a CDMA scheme. CDMA utilizes spread spectrum modulation scheme to spread the digital waveform of each user over

the entire frequency spectrum that is allocated for all users. Each user is separated by using a unique code. Finally, in order to use the frequency spectrum efficiently, the same carrier frequency can be reused in the different cells. This is the primitive form of SDMA. For more advanced forms of SDMA, the 120° sectorial beams at different carrier frequencies can be used within a cell and each sectorial beam can be used to serve the same number of users as are served in the ordinary case. The ultimate form of SDMA, however, is to use independently steered high-gain beams at the same carrier frequency to provide service to individual users within a cell. The latest form of SDMA usually employs an adaptive array antenna, which have recently drawn considerable interest from the communications community [1].

With recent rapid increase in demand for wireless communications as discussed above, in wireless communication systems, multipath fading effects and interference among communications clearly become serious problems. Normally, the severity of each problem depends upon a certain application under consideration. For example, in a direct sequence-code division multiple access (DS-CDMA) system, since the path delays for individual users usually span at most a few chips, intersymbol interference (ISI) due to the propagation delay through the multipath channel is relatively small and typically ignored [2]. In fact, performance degradation of the DS-CDMA system is mainly due to multiple access interference (MAI), which can be mitigated by a smart antenna. In general, one can classify the smart antenna into three categories, namely diversity, switch beam and adaptive antennas.

An adaptive array antenna is an array of antennas, which is able to change its antenna pattern dynamically to adjust to noise, interference, and multi-path. It can adjust its pattern to track portable users. Moreover, it is used to enhance received signals and may be used to form beams for transmission. Fig. 1.1 depicts the basic configuration of an adaptive array antenna. The signal received from each antenna element is multiplied by adjustable weights. The adjustable weights are obtained by using a specific adaptive algorithm. An adaptive algorithm is a procedure used to deal with the ability of a system to change its characteristics according with the condition which relies upon the selected adaptive algorithm. The adaptive algorithm can be applied into various applications such as adaptive filter, adaptive equalizer, adaptive antenna, etc.

The adaptive algorithm distinguished by a need for reference signal can be classified into two categories: non-blind and blind adaptive algorithms [3]. The non-blind adaptive algorithm is a computational method where its coefficient adjustment in the adaptive antenna relies upon a reference signal, which is often referred to as a training sequence. A least mean square (LMS), recursive least squares (RLS) and their derivatives are among adaptive algorithms commonly employed for the system [4]. They converge to the optimum Wiener solution [4]. In contrast, the blind adaptive algorithm does not require reference signal.

The first category of the blind algorithms is the Bussgang algorithm, the derivation of which is based upon maximum a *posteri* (MAP) estimation of the transmitted symbol, given the certain assumptions on its output. The examples of the Bussgang algorithm are the decision-directed algorithm, Sato algorithm, and CMA [3].The basic model of the Bussgang algorithm in an adaptive filter aspect is shown in Fig. 1.2. The Bussgang algorithm has the property that its autocorrelation function is equal to the cross-correlation between that process and the output of a zero-memory nonlinearity produced by the process, with both correlations being measured for the same lag.

During the last decade, the blind adaptive algorithm has received considerable interest for their application in wireless communication systems, see e.g. [3], because of their distinct advantage of not requiring the training sequence. One of the most popular blind algorithms is a constant modulus algorithm (CMA) [5], which can eliminate the interference with the condition that the transmitted signal has a constant envelope, e.g. a frequency- or phase-modulated signal. This is because the interference signal involves the amplitude fluctuation on the received signal. The CMA is blind since it employs only *priori* knowledge of the constant envelope of the transmitted signal to suppress the effect due to interference.

The algorithm was originally proposed in the context of blind equalization [5] and laterally applied to adaptive antenna systems where its performance was widely investigated, see e.g. [6]-[9]. Implementation of a CMA adaptive antenna for a Gaussian minimum shift keying (GMSK)-modulated mobile communication system was presented in [10] and its performance was reported in [11]. In [12]-[13], the CMA was used to reduce the co-channel interference in digital land mobile communications. A multistage constant modulus (CM) array comprising a multistage cascade of signal recovering and cancelling processes was proposed in [14]-[15]

where a sample matrix inversion (SMI) algorithm was applied to adjust the array weights of the signal cancelling method for bit error rate (BER) improvement. In [16], the CMA was modified to control a microwave analogue beamformer via a variable phase shifter. This algorithm was referred to as the M-CMA. In addition to reduction of co-channel interference, the CMA was exploited for signal recovery in fast fading environments [17].

So far, the main drawback of applying the CMA to the adaptive antenna system is its slow convergence property [13]-[19]. It was reported that the convergence behaviour of CMA heavily depends on its initial condition [18]-[20]. Recently, a solution to resolve this convergence problem was introduced by utilizing a solution of the SMI to initialize the CMA [21]. Nevertheless, there are two shortcomings in this approach. Firstly, inversion in the SMI implies a high computation burden of the algorithm. Secondly, the algorithm is not purely blind because it needs a reference signal.

In [22], the RLS-CMA algorithm was proposed to enhance the CMA convergence properties where the original CMA was converted to the new algorithm based on the analogy to an RLS optimization technique. Though the convergence rate is improved, there is a trade-off between computation burden and convergence performance. Therefore, both aforementioned algorithms are not suitable for hardware implementation.

By the way, in future wireless communication systems, the architecture which can operate at a high data rate is also expected to play an important role to fulfill its service. Practically, the recursive algorithm imposes a critical limit on its implementation. The algorithm can be generally implemented on both digital signal processor (DSP) and field programmable gate array (FPGA) [10], [13] and [23]. In these approaches, a multiplier is integrated with an adder such that multiply-accumulate (MAC) operation can be carried out in one clock cycle. It is noticed that it usually cannot operate at high clock frequency. To overcome the slow operating clock speed, two popular approaches namely pipelining and parallel processing were introduced to increase clock speed or sample speed [24]-[26].

In [27]-[30], pipelining technique was applied for hardware implementation of the LMS algorithm in order to achieve a high throughput. Modification of the original LMS algorithm was proposed to make the implementation by using pipelining technique possible. Two modified algorithms are the relaxed look-ahead pipelined

LMS (RLPLMS) and delayed LMS (DLMS) algorithms [27], [29]. In the DLMS algorithm, delay units D are inserted between the tap of an adaptive digital filter. This approach can increase throughput. However, the delay in the adaptation coefficient results in output latency and degrades the convergence property of DLMS algorithm. In [31], a high-speed FPGA implementation of the pipelined LMS filter based on the DLMS algorithm was presented for electronic support measures receivers. Note that there are no literatures relevant to the aspect of a high-speed hardware implementation of CMA for an adaptive antenna system.

Besides a high-speed architecture, the efficient arithmetic is required to reduce the cost and power consumption of the target device. A fixed-point arithmetic has been widely used to implement an adaptive processing unit. In [10], the CMA adaptive array was implemented on a DSP device using 16-bit fixed-point operations. The device can completely perform one sample in 2 ms. It means that the sampling rate is only 0.5 ksps. The pipelined adaptive LMS predictor proposed in [31] was also designed based on a fixed-point arithmetic. Another technique presented to reduce the complexity of multipliers is the power-of-two multiplication. The coefficients of the digital filter were restricted to power-of-two or the sum of two terms which are powers of two [31]-[35]. Moreover, a power-of-two quantizer was used at the input of the multiplier in an adaptive equalizer [36]. In these approaches, the multipliers can be implemented by simple shifters and adders.

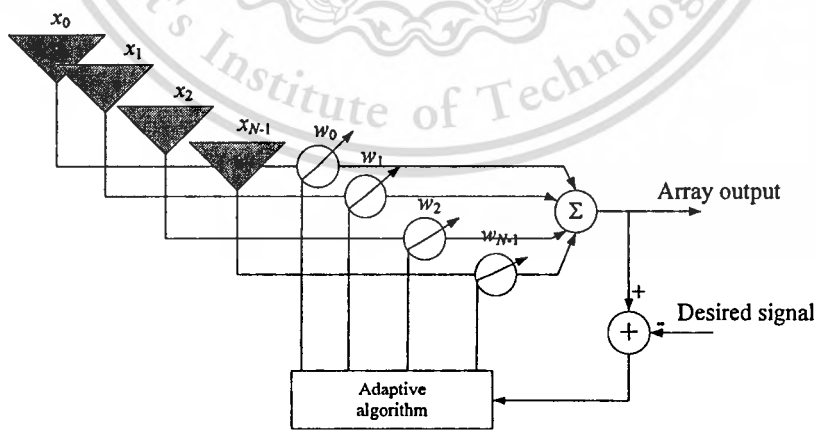


Fig. 1.1 Basic configuration of an adaptive array antenna.

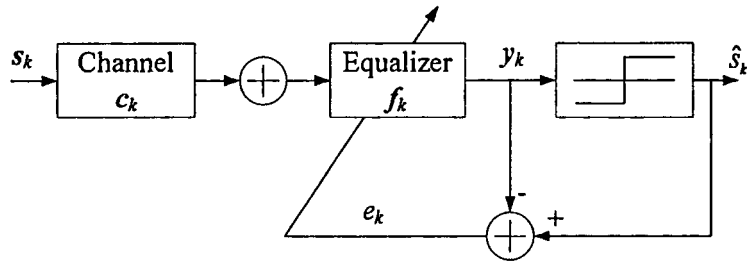


Fig. 1.2 Basic model of the Bussgang algorithm.

1.2 Research Objective

As aforementioned, the main drawback of applying CMA to an adaptive antenna is its slow convergence property. There are a number of publications relevant to the techniques which can resolve the underlying problem. However, they do not seriously take into account the aspects of implementation.

Currently, there exist a large number of techniques proposed for a multi-beam antenna, see e.g. [37]. In essence, the antenna elements are spatially arranged and electrically interconnected to form a phased array. Beam-switching (or beam-scanning) of the antenna is normally achieved via a phase shifter operating at either RF or IF.

Among the incoming beams to the receiving antenna, there is always a beam with maximum power. It is rational to utilize this beam for a CMA initialization. Demonstrating the concept in simulation [38], the system initially scans its beam and then determines a maximum received power beam, which is efficiently exploited for initialization of the adaptive antenna. Subsequently, the CMA can be used for further adjustment of the antenna beam. It was shown that the approach can significantly improve the convergence property of the CMA.

One of the goals of this thesis is to present an initialization technique which can improve convergence property of CMA by maintaining its advantage of not requiring a reference signal. The implementation aspects of the proposed concept will be addressed. It will be shown that the underlying initialization technique can easily be achieved using a simple hardware circuitry which only comprises a power detector and phase shifters. Furthermore, implementation of an adaptive processor and associated control unit using a field programmable gate array (FPGA) will be discussed. The performance of the FPGA implementation of the CMA adaptive antenna system with hardware-assisted initialization is investigated.

Despite the potential of CMA, another main drawback is that it has not been implemented efficiently on the target device. Thus, another goal of this thesis is to present the efficient architecture for hardware implementation of a CMA adaptive array antenna. The delayed CMA (DCMA) in context of the adaptive antenna system will be described while its solution will be determined also. The architecture of the pipelined CMA based on DCMA will be proposed along with investigation of the effect of its wordlenght. Fixed-point multipliers for multiplication between input signals and others are replaced by power-of-two multipliers in order to reduce the consumed resource for hardware implementation. The performance of the proposed architecture is evaluated via implementing it on FPGA.

1.3 Organization of the thesis

After introduction, the thesis is organized as follows. Chapter 2 provides the theoretical background of the circular phased array antenna, CMA array, and FPGA design. The derivation of update equations of CMA is given along with its illustrative example via simulations. The conventional CMA array is modified to be a CMA adaptive phased array by utilizing a simple circuit of phase shifters. The update equations of the modified CMA are also discussed via simulations with and without the present of noise. Moreover, pipelining technique used to improve a speed performance for implementation is addressed. Several ways to pipeline the multiply and accumulate (MAC) unit are introduced. Since the target hardware used to implement an adaptive antenna is FPGA, the basic architecture and design of FPGA is presented.

In Chapter 3, a cost-effective approach for initialization of the adaptive antenna using CMA is presented. The simulations were conducted to validate the proposed technique. The additional case studies of the proposed technique are given such as the case of two interference signals in order to investigate its behavior that the antenna can be used in the practical environment. Besides, the design of the adaptive processing unit using CMA is presented. The basic operators including the adders, subtractors, multipliers, and number converters in the unit are discussed. The unit is implemented on FPGA. Moreover, the antenna prototype is constructed and shown. Finally, experimental results with the flat four-beam compact phased array antenna

[37] being designed for the IMT2000 system of 1.95 GHz are also presented in comparison with the simulation results.

In Chapter 4, a new antenna, a switched-beam element phased array antenna, is used to replace a flat four-beam phased array antenna. The characteristic of the switched-beam element phased array antenna will be described in Section 4.2 where the adaptive antenna system along with the switched-beam element phased array antenna will be proposed. In Section 4.3, the simulations are conducted to show that the proposed adaptive antenna with the new antenna has the fasted convergence rate.

In Chapter 5, the DCMA in context of an adaptive array antenna will be introduced along with its signal model and update equations. The effect of inserting delay units into CMA is investigated via simulations with 2-element linear antenna. The characteristics of DCMA-based the proposed adaptive antenna are shown in Section 5.3.2. The efficient pipelined architecture based on DMCA is proposed in Section 5.3.3 in which fixed-point and power-of-two operations are utilized to reduce the consumed resource for hardware implementation.

In Chapter 6, the proposed pipelined architecture based on DCMA adaptive array is implemented on FPGA by using fixed-point and power-of-two operations. Fixed-point and power-of-two representations are described in Section 6.2. The basic component circuits used to implement the proposed pipelined architecture are discussed in Section 6.3. The testing system for the proposed pipelined architecture using fixed-point and power-of-two operations is shown. The implemented DCMA-based architecture is tested. The implementation results of the proposed pipelined architecture are shown validate that the consumed resource of the proposed DCMA-based architecture using power-of-two operations is less than that of the proposed DCMA-based architecture using fixed-point operations.

Finally, the conclusions and discussions are given in Chapter 7.

CHAPTER 2
THEORETICAL BACKGROUND AND RELATED WORKS

2.1 Introduction

As aforementioned in chapter 1, two efficient techniques for hardware implementation of an adaptive antenna using CMA will be presented. In this chapter, the theoretical background of a phased array antenna and CMA array is discussed. The derivation of CMA is presented along with its analysis through illustrative simulations. Moreover, the architecture for design of FPGA is also discussed and will be applied for efficient-hardware implementation in the next chapter.

2.2 Four-element Circular Phased Array

The applications of mobile communication system require that a main beam of the antenna can be changed or scanned. It can be accomplished by one of the efficient techniques of array along with fixed phase to its elements, i.e. a phased array. Generally, the phased array is defined as an array antenna whose main beam maximum direction is controlled by varying the phase or time delay to the elements. In this section, the concept of a four-element circular phased array antenna is presented. The parameters of the antenna are chosen to obtain the desirable pattern characteristic. Then, one-bit phase shifters applied to scan the beams are discussed.

The circular array comprises antenna elements which are placed on circle ring. The radiation pattern can be chosen by changing the phased excitation of the antenna elements. Fig. 2.1 depicts the basic configuration of a circular array.

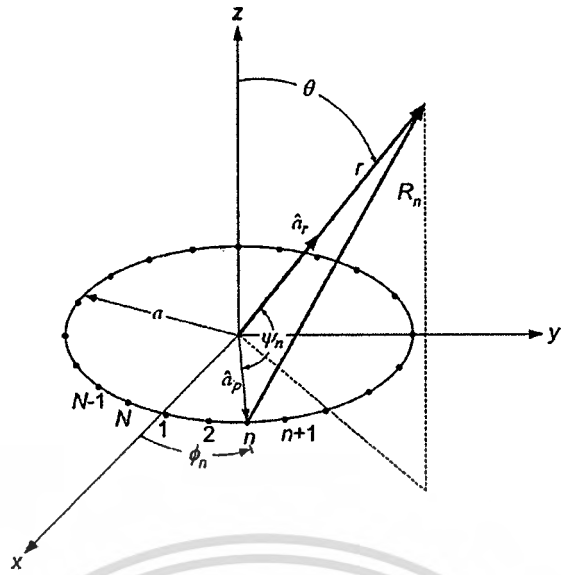


Fig. 2.1 Basic configuration of *N*-element circular array antenna.

Let us consider the array factor of the circular array antenna as seen in Fig. 2.1. The *N* isotropic sources are equally placed on an x-y plane with a circular radius *a*. Following the derivation in [39], the normalized electric field can be expressed as

$$E_n(r,\theta,\phi) = \frac{e^{-jkr}}{r}\left[\mathrm{AF}(\theta,\phi)\right] \tag{2.1}$$

where *k* is a propagation constant and AF is an array factor defined by

$$\mathrm{AF}(\theta,\phi) = \sum_{n=1}^{N} a_n e^{+j\left[ka\sin\theta\cos(\phi-\phi_n)+\alpha_n\right]} \tag{2.2}$$

where a_n is a excitation coefficient (amplitude) of the *n*th element, α_n is a phase excitation (relative to the array center) of the *n*th element, and $\phi_n = 2\pi\left(\frac{n}{N}\right)$ is the angular position of the *n*th element on the a x-y plane. *N* is the number of antenna elements.

Eq. (2.2) indicates that the main beam direction (θ_0,ϕ_0) can be controlled by choosing the phase excitation to the *n*th element as given by

$$\alpha_n = -ka\sin\theta_0\cos(\phi_0-\phi_n) \tag{2.3}$$

So far, the array factor in Eq. (2.2) can be rewritten as

$$\begin{aligned} \mathrm{AF}(\theta,\phi) &= \sum_{n=1}^{N} I_n e^{+j\left[ka\sin\theta\cos(\phi-\phi_n)-ka\sin\theta_0\cos(\phi_0-\phi_n)\right]} \\ &= \sum_{n=1}^{N} I_n e^{jk_a(\cos\psi-\cos\psi_0)} \end{aligned} \tag{2.4}$$

where I_n is a current of the nth element.

In [37], the phased array antenna was presented on the basis of the circular array theory along with the parameter conditions as follows.

1. The antenna was designed to provide the radiation pattern in the azimuthal plane. The peak beam direction θ_0 of 30° was defined to tilt the beam of the antenna for the specific coverage area.
2. The antenna had the beamwidth of 90° in four different directions.
3. The number of the antenna elements was selected to be four ($N = 4$)
4. The excited amplitude for each element was equal.
5. To provide the radiation pattern in azimuthal plane and obtain the desirable front-to-back ratio (F/B ≥ 10 dB) characteristic, the array radius a of the antenna must be considered.

By varying the array radius, Fig. 2.2 shows the radiation patterns of the main beam direction when $\phi_0 = 45°$. The main beam is relatively wide when $a = \lambda_0/4$ as seen in Fig. 2.2 (a). The F/B ratio is less than 3 dB. Fig. 2.2 (b)-(d) respectively illustrate the radiation pattern when $a = 2\lambda_0/4$, $3\lambda_0/4$ and $4\lambda_0/4$. These figures reveal that their half-power beamwidths are narrower than 90°. Thus, they are not sufficiently wide to cover the specific area (360°) when scanning the main beam in four directions. Besides, these are not suitable to provide the directional antenna because the side lobe is relatively high. As mentioned above, the main beam obtained from choosing $a = \lambda_0/4$ is wide. On the contrary, the main beam obtained from choosing $a = 2\lambda_0/4$ is relatively narrow while the back lobe is high. Therefore, the array radius of the antenna is chosen to be $3\lambda_0/8$ in order to increase the beamwidth as well as reduce the back lobe. The radiation pattern obtained from choosing $a = 3\lambda_0/8$ is illustrated in Fig. 2.3. The half-power beamwidth is about 120° which can cover the specific area. The F/B ratio is higher than 10 dB which can provide the directional radiation pattern.

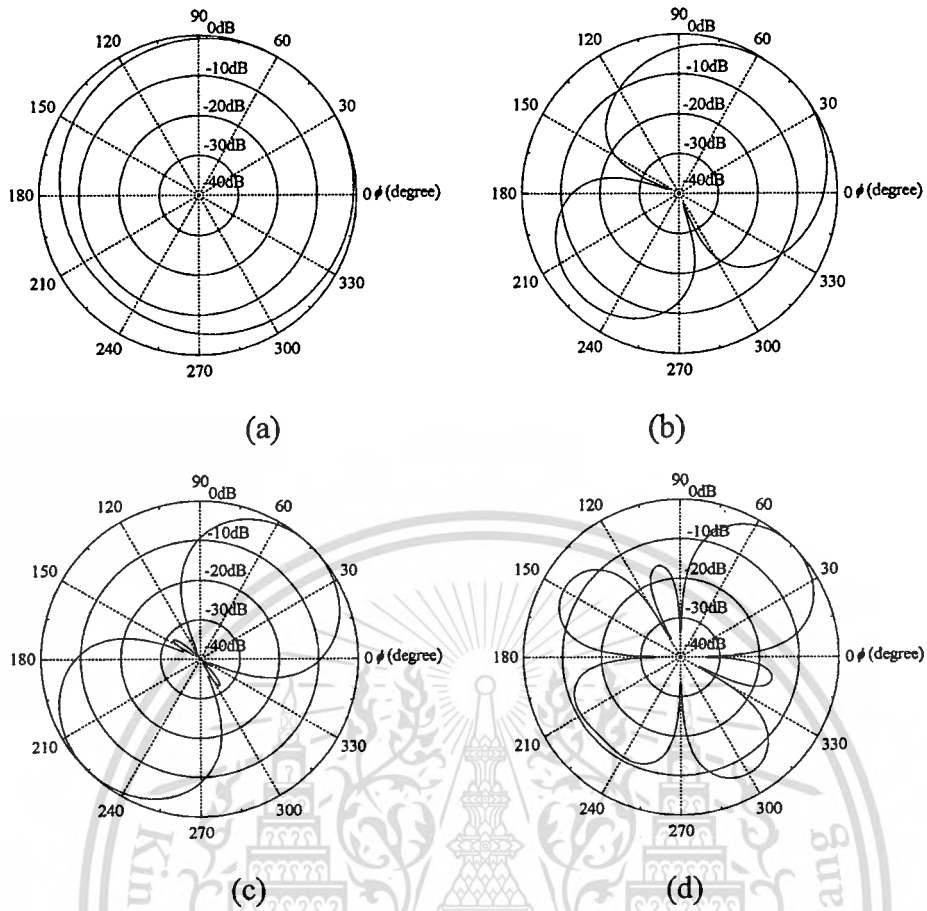


Fig. 2.2 Radiation patterns of the circular array antenna when a = (a) $\lambda_0/4$ (b) $2\lambda_0/4$ (c) $3\lambda_0/4$ (d) $4\lambda_0/4$.

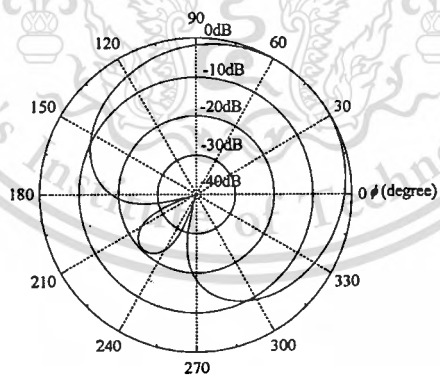


Fig. 2.3 Radiation pattern of the circular array antenna when $a = 3\lambda_0/8$.

In addition, the number of antenna elements was changed and the consequent radiation pattern was then considered. Fig. 2.4 (a)-(b) show the radiation pattern of the circular array antenna when $N = 2$ and 8, respectively. Fig. 2.4 (a) reveals that the main beam directs to 45° and 135° and the F/B ratio is less than 10 dB. The antenna with $N = 2$ has directional pattern with wide beamwidth and high back lobe and is therefore not

suitable to be employed for beam-switching. In Fig. 2.4 (b), despite no back lobe, there are slight differences in the radiation patterns between the cases of $N = 4$ and 8. The array with $N = 4$ is sufficient to obtain the F/B ratio above 10 dB. Thus, it is not necessary to increase the number of antenna elements because of the expensive cost.

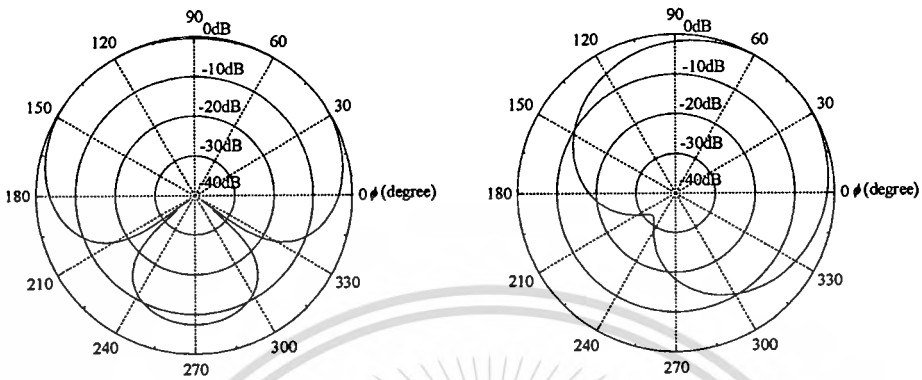


Fig. 2.4 Radiation pattern of the circular array antenna when N is (a) 2 (b) 8.

According to the parameters 1-5, they were substituted into Eq. 2.3 as

$$\begin{aligned}\alpha_n &= -\left(\frac{2\pi}{\lambda_0}\right)\left(\frac{3\lambda_0}{8}\right)\sin(30°)\cos\left(\phi_0 - \frac{\pi n}{4}\right) \\ &= -\frac{3\pi}{4}\sin(30°)\cos\left(\phi_0 - \frac{\pi n}{4}\right)\end{aligned} \tag{2.5}$$

Table 2.1: Phase shifters for four different beams

Phase / Beam Direction (ϕ_0)	Element 1	Element 2	Element 3	Element 4
45°	-47.7°	47.7°	47.7°	-47.7°
135°	-47.7°	-47.7°	47.7°	47.7°
225°	47.7°	-47.7°	-47.7°	47.7°
315°	47.7°	47.7°	-47.7°	-47.7°

By using Eq. (2.5), the phase excitations to obtain four different main beam directions of ϕ_0 = 45°, 135°, 225°, and 315° can be calculated. Table 2.1 lists phases for

realizing each beam. Note that the phase excitation can be achieved by four one-bit phase shifters of ±47.7° offering a cost-effective solution [37].

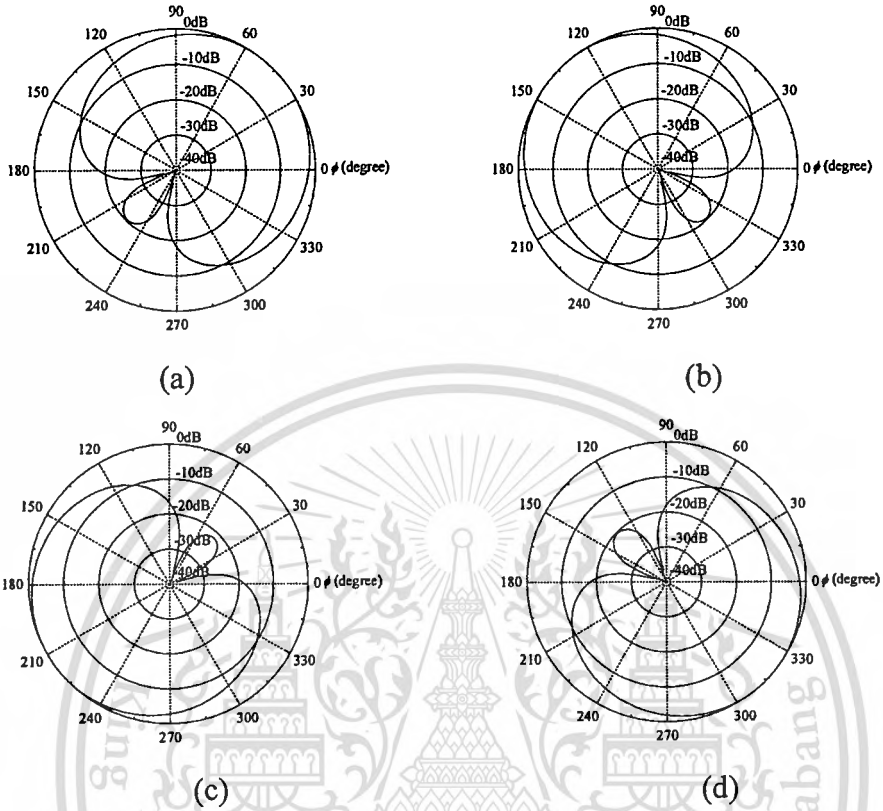


Fig. 2.5 Four different main beams direct to (a) 45° (b) 135° (c) 215° (d) 315°.

2.3 CMA Adaptive Array

The CMA was originally proposed to eliminate interference due to fluctuation in transmitted signal [5]. This algorithm does not require a training sequence and therefore is referred to as a blind adaptive algorithm. To eliminate interference, CMA exploits a known property of transmitted signal, i.e. its constant envelope. Examples of this type of signal include FM, QPSK, etc.

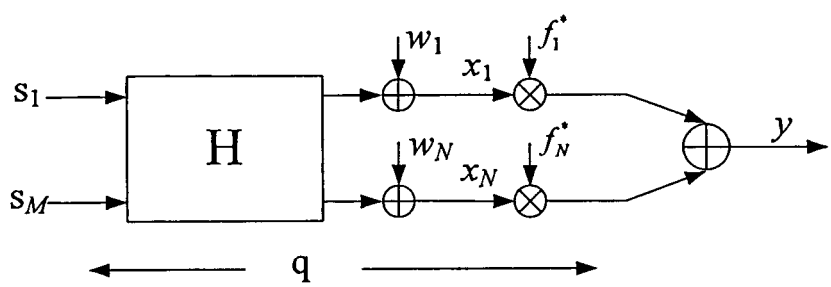


Fig. 2.6 The multi-input multi-output model.

2.3.1 The model and functional form

Constant modulus (CM) receiver can be applied into an adaptive antenna system. In this section, let us consider the estimation problem in the multiple-input multiple-output model shown in Fig 2.6. The received signal vector for CM receiver is given by

$$\mathbf{x} = \mathbf{H}\mathbf{s} + \mathbf{w}, \tag{2.6}$$

where $\mathbf{x} = [x_0, x_1, \ldots, x_N]^T \in \Im^N$, $\mathbf{H} \in \Im^{N \times M}$ is the (unknown) array respond matrix, $\boldsymbol{s} = [s_1, s_2, \ldots, s_M] \in \Im^M$ is the vector of source signals, and $\mathbf{w} = [w_1, w_2, \ldots, w_N] \in \Im^N$ is the additive noise. The linear estimate output y of CM receiver can be obtained by

$$y = \mathbf{f}^H\mathbf{x} = \mathbf{q}^H\mathbf{s} + \mathbf{f}^H\mathbf{w}, \tag{2.7}$$

where $\mathbf{f} = [f_1, f_2, \ldots, f_N]^T \in \Im^N$ is the estimator response vector, and $\mathbf{q} = \mathbf{H}^H\mathbf{f} \in \Im^M$ is the (conjugate) total system response.

In [5], the constant modulus criterion was proposed to minimize dispersion function defined by

$$J(\mathbf{f}) = E\{(|y|^2 - r)^2\} = E\{(|\mathbf{f}^H\mathbf{x}|^2 - r)^2\} \tag{2.8}$$

where $r = E\{|s_i|^4\} / E\{|s_i|^2\}$ is the dispersion constant. The signal s_i is to be estimated (desired signal). Similarly, the minimization can approximate to that of mean-squared error. Conventionally, when there is no noise, the CM receiver converges globally to the channel inverse which is the Wiener receiver.

Practically, a CM receiver is usually obtained from the stochastic gradient algorithm used as an iterative minimization of $J(\mathbf{f})$. That is

$$\mathbf{f}(k+1) = \mathbf{f}(k) - \mu \frac{\partial}{\partial \mathbf{f}} J(\mathbf{f}) \tag{2.9}$$

where μ is a small step size governing convergence rate and $\frac{\partial}{\partial \mathbf{f}} J(\mathbf{f})$ is an unbiased estimate of the gradient of the cost function with respect to $\mathbf{f}(k)$. The gradient of $J(\mathbf{f})$ is given by

$$\frac{\partial}{\partial \mathbf{f}} J(\mathbf{f}) = 2E\{(|\mathbf{f}^H \mathbf{x}|^2 - r)^2 y^* \mathbf{x}\} \tag{2.10}$$

By substituting Eq. (2.10) into Eq. (2.9) and removing the expectation operation, it is easily verified that

$$\mathbf{f}(k+1) = \mathbf{f}(k) - \mu \mathbf{x}(k) y^*(k)(|y(k)|^2 - r)^2 \tag{2.11}$$

Considering the general case, the following assumptions can be applied.

A1: $\mathbf{H} \in \Im^{N \times M}$ has full column rank.

A2: $\mathbf{w}$ is Gaussian with zero mean and covariance $\sigma \mathbf{I}^2$, and independent from $\mathbf{s}$.

A3: Source vector $\mathbf{s}$ is sub-Gaussian with independent components.

A source is super- (sub-) Gaussian when $\kappa_s < \kappa_g$ ($\kappa_s > \kappa_g$) and it satisfies kurtosis κ_s defined by

$$\kappa_s \triangleq \frac{E\{|\mathbf{s}|^4\}}{E\{|\mathbf{s}|^2\}^2} \tag{2.12}$$

where the kurtosis of a real-valued Gaussian random process is $\kappa_s = 3$ and that of a proper complex-valued Gaussian random process is $\kappa_s = 2$. Assumption A1 implies that the number of antennas is no less than that of sources and there are no sources propagating in the same path. In case of no additive channel noise, assumption A1 can pertain to the channel equalizer pair's ability to achieve perfect equalization [40]-[41]. Assumptions A3 pertain to only blind equalization based on the CM receiver.

2.3.2 Illustrative examples of CM dispersion function and its solution

Referring to Eq. (2.8), the CM dispersion function can be illustrated in Fig. 2.7 as a function of the output signal space in the real and imaginary components. Here, r is defined to be one. As seen in the figure, the function J is minimized to be zero on a unit-radius circle with center at origin. The contour of $|y| = 1$ is desired for convergence of CMA. When the algorithm converges to this situation, its adaptation will be stopped. This results from the algorithm which is based on the gradient method as defined in Eq. (2.11). Note that the zero-gradient condition can be occurred when $|y| = 0$. However, [42] mentioned that this is not a practical solution with two given reasons. Firstly, with the present of noise or other inducements, $\mathbf{f}$ will be moved away from the zero-gradient point. Thus, the point of $|y| = 0$ is an unstable equilibrium. Secondly, since the input signal $\mathbf{x}(k)$ normally has a nonzero bandwidth, $\mathbf{f} = 0$ is only condition to give a zero-gradient point. Note that $\mathbf{f} = 0$ is realized when only starting up an adaptation of the algorithm. Traditionally, all-zero initial vector of $\mathbf{f}$ should not be employed. In addition, they showed that CMA which is initialized by using unit-spike initialization (N-1 zeros and one unity gain) provide an initial all-pass response.

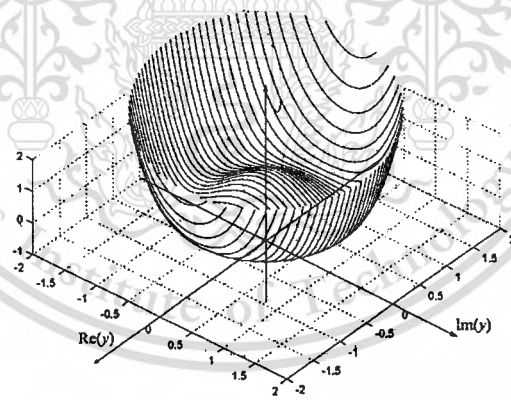


Fig. 2.7 CM dispersion function as a function of the output signal space.

In [43], it was shown that all CM local minima inverse the channel matrix for the equalization problem. When H is square and full rank (assumption A1), the optimization of $\mathbf{f} \in \mathfrak{I}^N$ is in general equivalent to the optimization of the system response defined by

$$\mathbf{q} = \mathbf{H}^H\mathbf{f} \tag{2.13}$$

Thus, most existing analyses transform the optimization of receiver coefficients to the optimization of $\mathbf{q}$ [44]. The system responses of $\mathbf{q}=[\pm 1 \quad 0]^T$ or $\mathbf{q}=[0 \quad \pm 1]^T$ will be occurred for the zero-forcing (ZF) system [3].

To study convergence of CMA to its solution, let us first consider the array antenna comprising two omni-directional antennas ($N = 2$) with half-wavelength spacing. Moreover, desired and interference ($M = 2$) signals are independently $\pi/4$-QPSK-modulated and their incident angles are 30° and 120°, respectively. With the reference to the model, the array respond matrix $\mathbf{H}$ is

$$\begin{bmatrix} 1 & 1 \\ -0.91+0.41\mathrm{i} & -1.00\mathrm{i} \end{bmatrix} \tag{2.14}$$

Simulations were conducted to determine CMA behaviors. The power of the desired signal was set to be equal to that of the interference. The step size and the initial weight vector for performing CMA were 2^{-9} and $[1 \ 0]^T$, respectively. The simulations were divided into two cases: noiseless and noise cases.

Noiseless case:

There was no noise in this case. CMA was performed along with the condition as mentioned above. The weight vectors after convergence (at 35,000 iterations) of the antenna was $[0.57+0.16\mathrm{i} \ -0.16+0.57\mathrm{i}]^T$. This results in the overall response vector $\mathbf{q}$ being $[0.95+0.30\mathrm{i} \ 0.00+0.00\mathrm{i}]^T$. Equivalently, it corresponds to $\mathbf{q}$ being $[1e^{j0.31} \ 0e^{j0}]$ in polar form and clearly implies that the desired signal is perfectly captured with the overall phase offset being 17.7°. Fig. 2.8 illustrates the antenna pattern obtained from CMA after convergence. As seen in the figure, the direction of main beam directs to $\phi_0 = 60°$ and the nullity appears at $\phi_i = 120°$. The figure reveals that interference is completely eliminated and the system respond is almost occurred for ZF system [3].

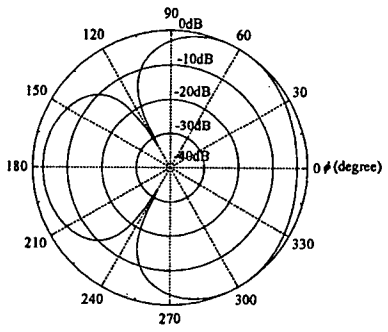


Fig. 2.8 Antenna pattern obtained from CMA after convergence with no noise.

Noise case:

In this case, the additive Gaussian noise of 20 dB of SNR was presented. The antenna parameters were identical to that in noiseless case. The weight vectors after convergence (at 80,000 iterations) of the antenna was $[0.60+0.01i \quad 0.55-0.23i]^T$, resulting in $\mathbf{q} = [0.00+0.00i \quad 0.83-0.56i]$ corresponding to $\mathbf{q} = [0e^{j0} \quad 1e^{-j0.59}]$ in polar form. The overall phase offset was -33.98°. In this case, the antenna captures interference rather than desired signal. The antenna pattern obtained from CMA after convergence is shown in Fig. 2.9. The direction of main beam directs to $\phi_0 = 226°$ and the nullity appears at the desired signal direction of 30° as seen in Fig. 2.9. This implies that the present of noise will disturb CMA behavior.

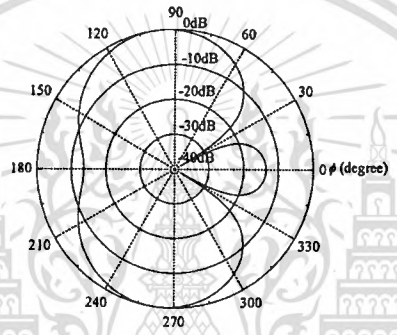


Fig. 2.9 Antenna pattern obtained from CMA after convergence when noise is presented.

2.4 CMA Adaptive Phased Array

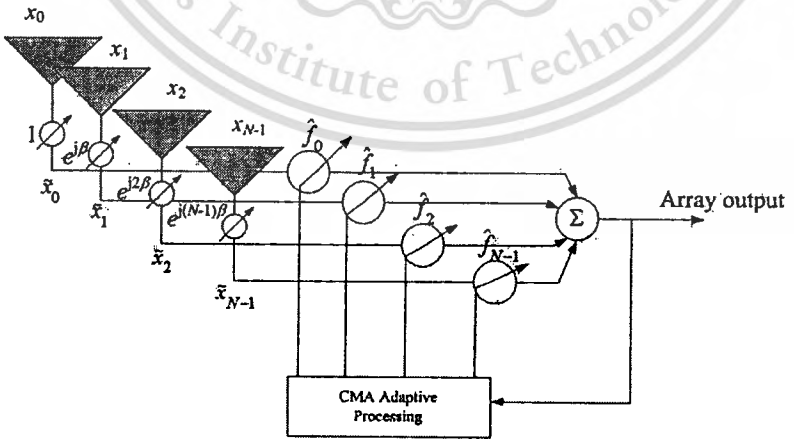


Fig. 2.10 Basic architecture of CMA adaptive phased array.

In this section, the concept of CMA adaptive phased array are discussed and then validated through simulations. Fig. 2.10 depicts a basic architecture of CMA adaptive phased array. A complex signal received by the i^{th} element of the antenna array is phase-excited. After phase excitation, an effective input of the CMA antenna system $\tilde{x}_i(k)$ is then multiplied by an adjustable complex weight $\tilde{f}_i(k)$, where k denotes the time index. The antenna output $y(k)$ is constructed by combining the product $\tilde{f}_i(k)\tilde{x}_i(k)$ as

$$y(k) = \sum_{i}^{N-1} \tilde{f}_i(k)\tilde{x}_i(k), \tag{2.15}$$

or more compactly

$$y(k) = \tilde{\mathbf{f}}^H(k)\,\tilde{\mathbf{x}}(k) \tag{2.16}$$

where $\tilde{\mathbf{f}}(k) = [\tilde{f}_0(k), \tilde{f}_1(k),\ldots, \tilde{f}_{N-1}(k)]^T$ and $\tilde{\mathbf{x}}(k) = [\tilde{x}_0(k), \tilde{x}_1(k),\ldots, \tilde{x}_{N-1}(k)]^T$ denotes a weight and effective input vector, respectively. Besides, N is the number of antenna elements. Without loss of generality, let us assume that the transmitted signal amplitude is unity, CMA adjusts $\tilde{\mathbf{f}}(k)$ in a way that the constant modulus (CM) cost function defined as

$$J(\tilde{\mathbf{f}}(k)) = \{[\tilde{\mathbf{f}}^H(k)\tilde{\mathbf{x}}(k)]^2 - 1\}, \tag{2.17}$$

is minimised. This results in the weight vector being recursively adjusted according to [5]

$$\tilde{\mathbf{f}}(k+1) = \tilde{\mathbf{f}}(k) - 4\mu\tilde{\mathbf{x}}(k)y^*(k)(|y(k)|^2 - 1) \tag{2.18}$$

where μ is a step-size.

Notice that the architecture described above slightly differs from the conventional CM array, in which the complex baseband signal $x_i(k)$ is directly fed to the adjustable gain element $\tilde{f}_i(k)$.

In [44], the constant modulus (CM) cost function was analyzed for array signal processing. It was shown that, when there is no noise and all sources are sub-Gaussian, all local minima of CM cost function are global and they perfectly estimate all the sources.

However, there is a main problem of the approaches when CMA is applied into beam-forming context. The extracted signal depends upon the initialization of CMA. Usually, the signal is unknown and extraction of signals is not certain. Thus, the appropriate initialization is necessary.

The relationship between the phased array antenna and CMA is illustrated by the following example. Here, the array antenna comprising two omni-directional antennas (N = 2) with half-wavelength spacing is considered. Moreover, desired and interference signals are $\pi/4$-QPSK-modulated. Fig. 2.11 shows the optimal pattern in Mean Square Error (MSE) sense obtained from the Wiener solution [45]

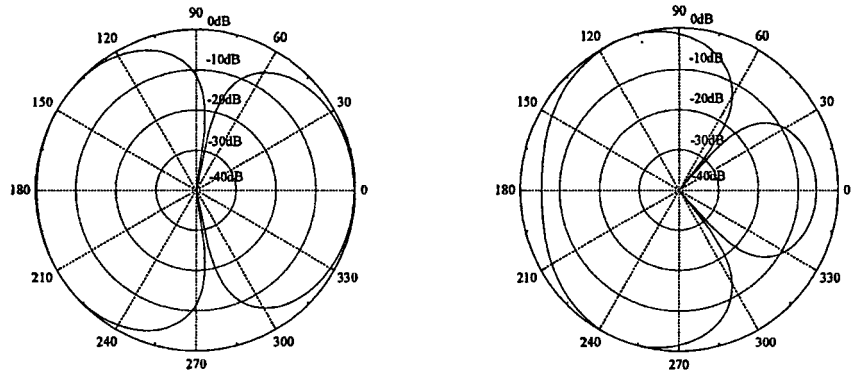
$$\mathbf{w}_{opt} = \mathbf{R}^{-1}\mathbf{s}^{*} \tag{2.19}$$

where * denotes conjugate, **R** is a covariance matrix and **s** is an invariant vector.

The example is distinguished into two cases. In case 1, desired and interference signals arrived in the direction of $\phi_d = 50°$ and $\phi_i = 80°$, respectively. By using eq. (2.19), the optimal pattern is shown in Fig. 2.11 (a) in which the nullity appears at the interference direction of 80°. In case 2, the incident angles of desired and interference signals were 80° and 50°, respectively. Fig. 2.11 (b) shows the pattern obtained from the Wiener solution. It is seen that now the nullity appears at the interference direction of 50°.

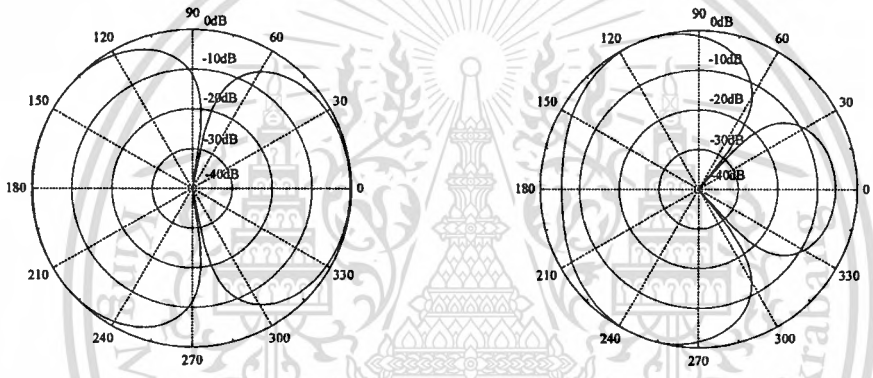
For adaptive algorithm scenarios, a main beam direction (ϕ_0) was switched to 50° for CMA beam initialization. This is attended by choosing a progressive phase of β = -115.7°. Fig. 2.12 (a) shows the pattern obtained from CMA beam-forming for case 1 after convergence. The nullity appears in the interference direction. Note that the pattern attained by the Wiener solution is in a perfect agreement with CMA beam-forming depicted in Fig. 2.11 (a). By switching the initial beam to $\phi_0 = 80°$ (choosing $\beta = -31.3°$), the pattern of Fig 2.12 (b) is obtained. This again agrees with the beam achieved from Wiener solution shown in Fig. 2.11 (b).

Further investigation reveals that, if the CMA is arbitrarily initialized (without beam-switching), an identical receiver is obtained for both cases. In other words, while one receiver can correctly capture the desired signal, the other receiver will rather capture the interference. This simple example confirms the need for appropriate CMA initialization.



(a) (b)

Fig.2.11 Antenna patterns obtained from Wiener solution when (a) $\phi_d = 50°$ and $\phi_i = 80°$ and (b) $\phi_d = 80°$ and $\phi_i = 50°$.



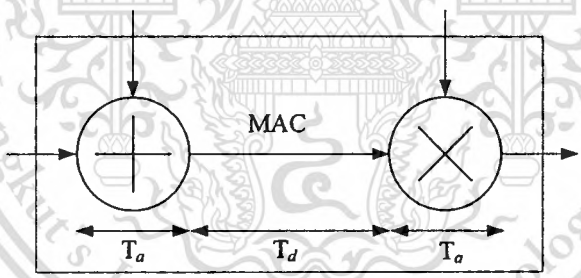
(a) (b)

Fig.2.12 Antenna patterns obtained from CMA with beam-switching when (a) $\phi_d = 50°$ and $\phi_i = 80°$ (b) $\phi_d = 80°$ and $\phi_i = 50°$.

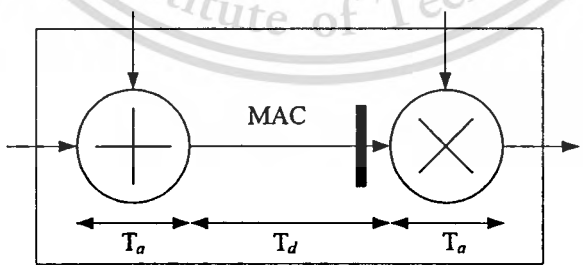
2.5 Pipelining

In real-time applications, the adaptive antenna system requires a high-throughput rate. To achieve a high-throughput rate, the critical path (the minimum time required for processing a new sample) should be reduced. It can be achieved by parallel processing or pipelining. Parallel processing is a technique for implementation to increase the sampling rate by replicating hardware so that several inputs can be processed in parallel and several outputs can be produced at the same time. Pipelining is used to reduce the effective critical path by introducing pipelining latches along the critical data path.

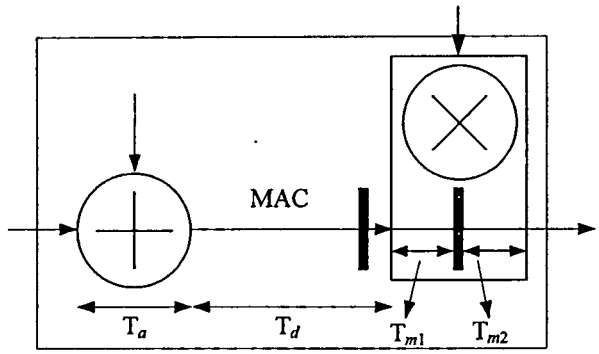
In chapter 4, the pipelining will be introduced to improve the performance of the CMA processing unit. To explain the idea of pipelining technique, let us consider the multiply and accumulate (MAC) operations which is in general used to be a main function unit in an operation for the adaptive filter and antenna. Pipelining of the MAC is not only limited to the adder and multiplier but also extended to the interconnect delay between these two components. The various ways to pipeline the MAC unit is shown in Fig. 2.13. Here, it is assumed that $T_m = 3T_a$ and $T_d = 1.5T_a$ [31] where T_m and T_a is respectively the processing time for a multiplier and adder. T_d is the interconnect delay between two components. The MAC unit without pipelining as seen in Fig. 2.13 (a) indicates that the critical path is equal to $T_a + T_m + T_d = 5.5T_a$. With pipelining, the MAC unit is shown in Fig. 2.13 (b)-(e). For example, the register is inserted between a multiplier and an adder as seen in Fig. 2.13 (b) and then the critical path is $T_m = 3T_a$. This results in one of latency. Despite the advantage, there are two main drawbacks in applying pipelining technique. Firstly, pipelining technique produces the latency in the designed circuit. Secondly, inserting the delay unit implies that number of delay elements (registers/latchs) in the critical path is increased.



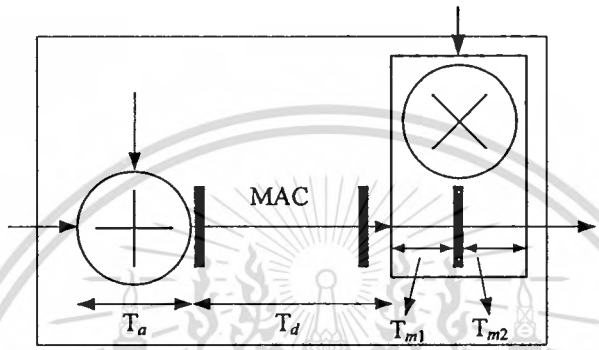
(a) Critical path = $5.5T_a$; Latency = 0



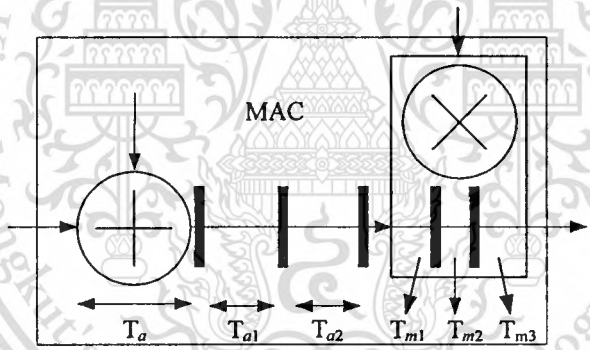
(b) Critical path = $3T_a$; Latency = 1



(c) Critical path = $2.5T_a$; Latency = 2



(d) Critical path = $1.5T_a$; Latency = 3



(e) Critical path = T_a; Latency = 5

Fig. 2.13 Pipelined MAC structure.

2.6 FPGA Technology

The programmable logic array was introduced in 1970. Subsequently, in 1985, Xilinx first introduced field programmable gate array (FPGA) defined as high density application specific integrated circuits (ASIC). Currently, FPGA devices, one of the modern technologies, are changing the electronic industry. They are now common in low and mid volume embedded products where they offer the following advantages:

-Fast time to market.

-Better integration.

-In system programmability.

-FPGAs tend to have long life cycles and are usually replaced by pin compatible parts.

-Cores such as PCI are available and able to integrate with relative ease.

FPFA can combine the logic integration of custom very large scale integration (VLSI) with the time of marketing and cost benefits of stand products.

As growing of semiconductor industry in the last decade, FPGAs such as Xilinx Virtex II family have grown in capacity up to 10 millions system gates. Moreover, Speed and function is also improved. The advantages of FPGA encourage designers to implement digital signal processing for operation in real-time. With these merits, FPGA is widely employed for design and implementation of adaptive algorithms.

2.6.1 FPGA architecture based on Xilinx Virtex-E family

In this section, the fundamental architecture of FPGA is described in order to comprehend a designed basic unit of an adaptive processor. Nowadays, FPGAs in the market are classified into two categories: SRAM-based FPGAs and antifuse-based FPGAs. Xilinx and Altera are leaders to produce FPGAs for the first category. For antifuse-based products which include Actel, Quicklogic and Cypress, and Xilinx offer to release their products into the competition. Here, SRAM-based FPGA which is the Xilinx Virtex-E family is discussed and then applied to design the CMA processing unit which is discussed in chapter 4 and 5.

The basic structure of Xilinx FPGA based on array is depicted in Fig. 2.14, comprising a two-dimensional array of logic blocks interconnected via horizontal and vertical routing channels using the programmable interconnections. Logic blocks provide the functional elements to construct the designer's logic. Input/ Output (I/O) blocks are employed to provide the interface between package pins and the logic blocks.

A logic block called a configurable logic block (CLB) of Xilinx Virtex-E family is shown in Fig. 2.15 [46]. A CLB is consisted of a four-input function generator, carry logic, and storage element. The function generators of Xilinx Virtex-E family are implemented as 4-input look-up tables (LUTs). Each LUT can provide a 16 x 1-bit synchronous RAM. Furthermore, the two LUTs within a slice can be combined to create a 16 x 2-bit or 32 x 1-bit synchronous RAM, or a 16 x 1-bit dual-port synchronous RAM.

The Virtex-E LUT can also provide a 16-bit shift register that is ideal for capturing high-speed or burst-mode data. This mode can also be used to store data in applications such as digital signal processing.

The storage elements in the Virtex-E slice can be configured either as edge-triggered D-type flip-flops or level-sensitive latches. The D inputs can be driven either by function generators within the slice or directly from slice inputs, bypassing the function generators. In addition to clock and clock enable signals, each slice has synchronous set and reset signals (SR and BY). The SR forces a storage element into the initialization state specified for it in the configuration. The BY forces it into the opposite state. Alternatively, these signals may be configured to operate asynchronously. All control signals are independently invertible and shared by the two flip-flops within the slice.

The F5 multiplexer in each slice combines the function generator outputs. This combination provides either a function generator that can implement any 5-input function, a 4:1 multiplexer, or selected functions of up to nine inputs. Similarly, the F6 multiplexer combines the outputs of all four function generators in the CLB by selecting one of the F5-multiplexer outputs. This permits the implementation of any 6-input function, an 8:1 multiplexer, or selected functions of up to 19 inputs. Each CLB has four direct feed through paths, one per LC. These paths provide extra data input lines or additional local routing that does not consume logic resources.

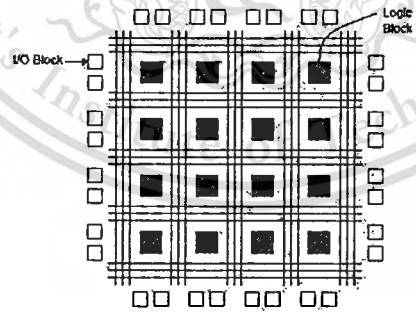


Fig. 2.14 Basic structure of Xilinx FPGAs based on array.

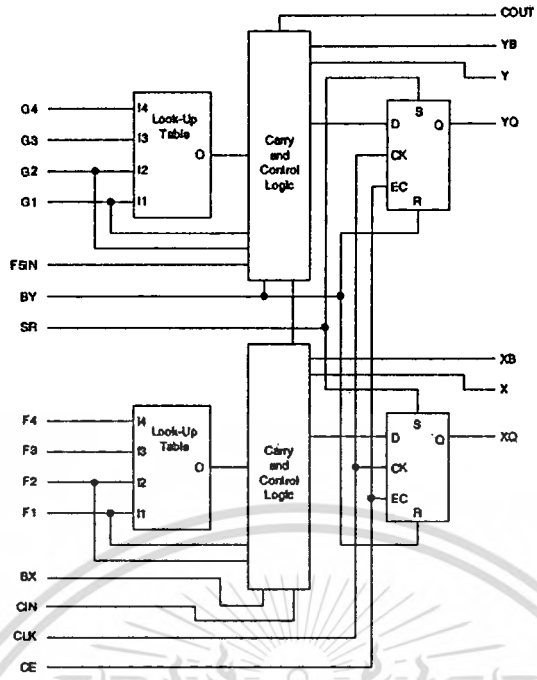


Fig. 2.15 CLB of Xilinx Virtex-E family.

2.6.2 FPGA design

This section shows the design methodology used to implement the FPGA-based adaptive antenna from algorithmic specifications. The methodology is divided into two domains: an algorithmic domain and an architectural domain. In this thesis, two techniques which can improve the convergence property and designed architecture of CMA are presented. To improve convergence, initialization technique utilizing beam-switching of the antenna will be introduced in the next chapter. To improve the designed architecture of the CMA processing unit, it is optimized to achieve the hardware which can operate at a high frequency clock and will be then discussed in chapter 4 and 5.

So far, let us consider the design flow of the adaptive algorithm. After specifying what the adaptive algorithm is, it was simulated along with the specified environment under our consideration. Using floating-point operation, this was conducted to determine the key parameters such as step size and the number of antenna elements. Those parameters were varied to achieve the desirable characteristic of the algorithm. The pipelined version of the algorithm can be re-simulated under the same environment to investigate the impact of pipelining to the algorithm. The floating-point operation was then converted to fixed-point operation to reduce the resource which was required for hardware implementation. The numbers of bits of input and output are chosen based on

their application. The wordlength of weights of the algorithm was minimized. The simulation results obtained from a high language description are used to validate the post-layout simulation obtained from the FPGA implementation process.

After simulating the algorithm with its key parameters, the architectural version of the algorithm should be carefully chosen or designed to reduce the critical path, hardware resources and output latency of the system. A very high description language (VHDL), based on the knowledge of the component size, performance, and the predictable speed and area of the optimized operator, is used to design the architecture of the algorithm. Fig. 2.16 shows the general flow for the HDL-based design.

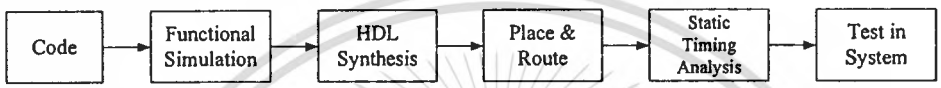


Fig. 2.16 Digital design flow.

There are a number of papers which were introduced to implement the hardware of the adaptive antenna, equalizer and filter. In [13] and [23], a digital beam forming (DBF) and beam space using CMA were presented to reduce interference and then implemented on FPGA. The processing unit of CMA was based on a MAC operation. All of functions for performing CMA require 10 FPGAs and can run at the frequency of 7.04 MHz. Note that with limited gate capacity, the unit cannot operate at the high frequency clock. In [10]-[11], the CMA adaptive antenna of a four-element antenna was proposed by using DBF concept for GMSK mobile communications. CMA was performed by two digital signal processors (DSP) with high computational burden.

In addition to CMA, LMS was widely used in scenario of the adaptive antenna and filter. In [47], the complexity for implementation of LMS was reduced by using a switched architecture. Following the LMS update equation and the approach [47], the computational complexity can be decreased to two complex multiples and one complex addition. In this approach, despite an additional switch, only a single analog-to-digital converter is required instead of one at each element. However, the main drawback of this approach is its decreased convergence rate.

To realize the adaptive antenna, it is noticed that not only the specific algorithm which should be efficient is necessary but also the signal processor with high performance make the adaptive antenna possible.

2.7 Summary

In this chapter, the theoretical background and related works have been discussed. The key parameters of the four-element circular phased array were investigated and chosen to obtain the directional pattern. The CMA adaptive array was discussed along with its model and derivation. The examples of CMA function in case of the presence of noise and noiseless were illustrated via simulations to investigate its behaviors. The pipelining technique which can improve the speed performance was explained because it will be applied to propose the efficient architecture introduced in Chapter 4. Moreover, FPGA which is a target device of this thesis was discussed.

CHAPTER 3
HARDWARE-ASSISTED INITIALIZATION FOR CMA ADAPTIVE ARRAY ANTENNA

3.1 Introduction

This chapter presents a cost-effective approach for initialization of the adaptive antenna using the constant modulus algorithm (CMA). The technique restores to one-bit phase shifters and a power detector which are normally integrated with a multi-beam antenna to determine a beam with maximum received power. Consequently, the beam is exploited as an initial beam for CMA. Development of hardware-assisted initialization is discussed. In addition, FPGA implementation of a CMA processor and associated control circuitry is presented. Simulations are shown to evaluate the performance of the proposed initialization technique. Besides, the developed prototype is tested with a phase array antenna designed for operation at IMT2000 frequency. Experimental results confirm superiority of the proposed technique.

3.2 A Flat Four-beam Phased Array Antenna

In [37], a flat four-beam compact phased array antenna was proposed for DCS-1800 at the frequency of 1.8 GHz. In this paper, the antenna was modified for operation at the frequency of 1.95 GHz suitable for indoor base station of the IMT2000 systems [48]. Fig. 3.1 shows a configuration of the flat four-beam compact phased array antenna. Beam-switching of the antenna can be achieved by utilizing one-bit phase shifters offering a cost-effective solution. Following the derivation in [37] and the previous chapter, the array radius "r_a" of a four-element circular array was chosen as 0.375λ in order to obtain the desirable characteristics.

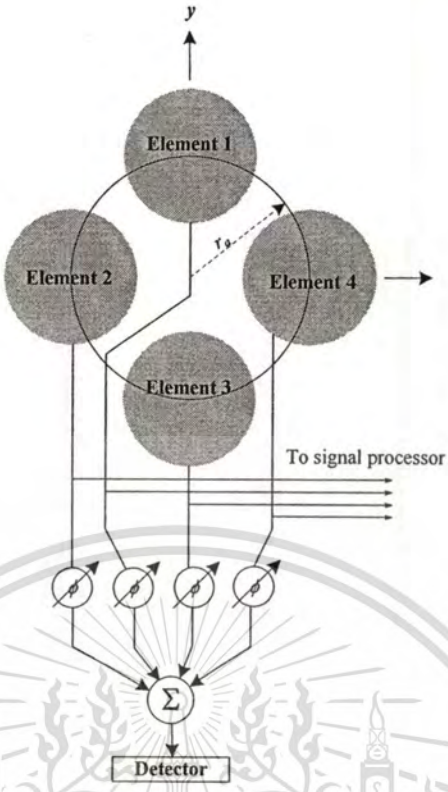


Fig. 3.1 Flat four-beam compact phased array antenna configuration.

3.2.1 Aperture-coupled circular patch antenna

The antenna design is based upon circular array principle. The circular microstrip antenna for the TM_{210} mode was designed to provide omni-directional pattern in azimuthal plane [39]. The peak beam direction θ_0 of 30° was defined to tilt the beam of the antenna for the specific coverage area. The circular patch elements were used to form the flat and compact structure.

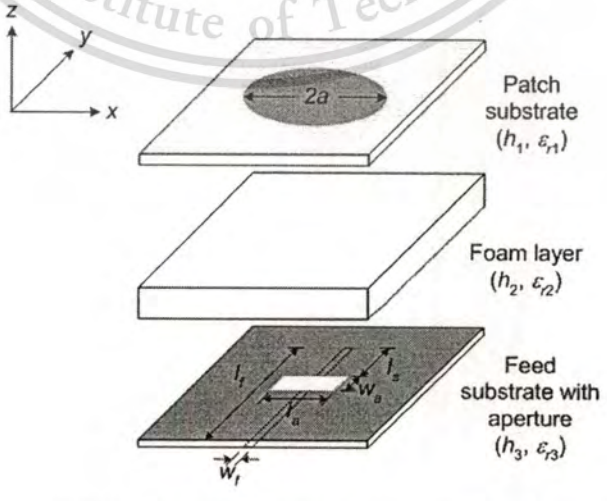


Fig. 3.2 Aperture-coupled circular patch antenna configuration.

A configuration of the aperture-coupled circular patch antenna is shown in Fig. 3.2. The circular patch with a diameter "*2a*" of 34 mm was etched on an FR-4 substrate with a height "h_1" of 1.5 mm and with a dielectric constant "ε_{r1}" of 4.3. The foam layer whose dielectric constant "ε_{r2}" is very close to that of the air was inserted between the patch substrate and the feed substrate to enhance the antenna bandwidth as well as to provide structural support. The thickness of the form layer "h_2" is 15 mm. The RO3003 material was employed for the feed substrate with a dielectric constant "ε_{r3}" of 3.0 and a thickness "h_3" of 0.75 mm. By varying the parameters of feed structure for optimal return loss, aperture width "w_a" of 5.33 mm, aperture length "l_a" of 53.30 mm, feed line width "w_f" of 2.00 mm and length of open circuited stub "l_s" of 3.55 mm were selected.

3.2.2 One-bit phase shifter

A phase shifter was used to realize the phase excitation of each antenna element for desirable beam direction. According to Table 2.1, one-bit phase shifters of ±47.7° can provide switching of the main beam to four directions, *i.e.* at ϕ_0 = 45°, 135°, 225°, and 315°. The main beam in each direction possesses over 90° half-power beamwidth as shown in chapter 2. Thus, the antenna can cover overall azimuthal directions. Front-to-back (F/B) ratio of more than 10 dB is also achieved.

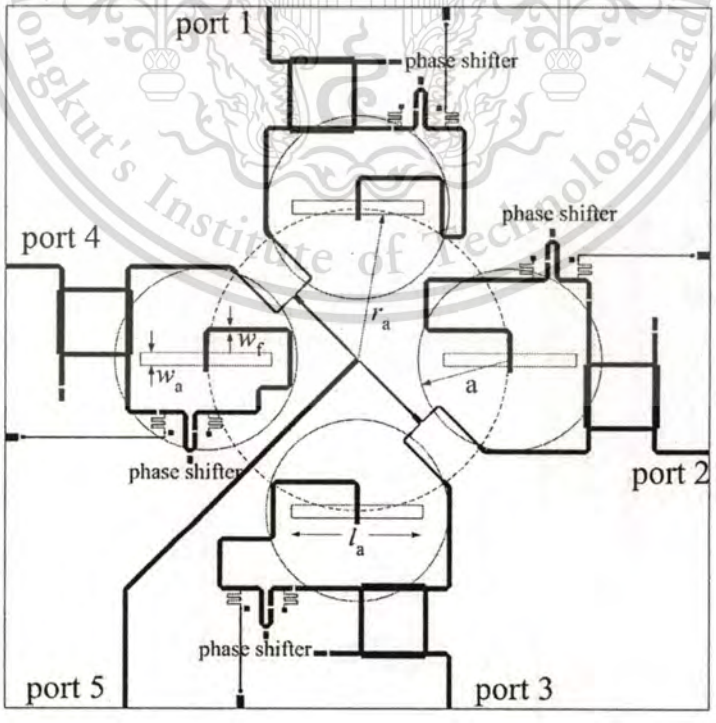


Fig. 3.3 Flat four-beam compact phased array antenna layout.

The layout of a flat four-beam compact phased array antenna is shown in Fig. 3.3. A series-and shunt-mounted 2-diode switched delay line phase shifter was applied for a one-bit phase shifter. The series diode was directly connected to input and output ports. The shunt diode was placed at the middle of the delay line. All was designed on the Rogers RO3003 PCB with a dielectric constant of 3.0 and a thickness of 0.75 mm to provide 47.7° phase shift at the designed frequency of 1.95 GHz. The RF choke was accomplished in the form of a thin zig-zag line of length $\lambda/4$, and the lumped chip capacitors were used as blocking capacitors. Two Philips BAP51-02 PIN diodes were used for switching the delay line and matching circuit.

In addition, according to [49], the branch-line coupler (port 1-4), 3 dB directional coupler with 90° phase difference in the outputs of the through and coupled arms, was added to couple the received signal for the signal processing unit while port 5, two-stage T-junction power combiner, is the array output port. VSWR is in the range of 1.5:1 for all ports in designed frequency band of 1.92-1.98 GHz. At operating frequency of 1.95 GHz, VSWR of port 1-5 measured by an HP8510C network analyzer were 1.26, 1.17, 1.24, 1.18 and 1.45, respectively. Finally, the antenna has linear polarization along the z-axis.

3.3 Hardware-assisted Initialization for CMA Adaptive Antenna

In a multi-beam antenna, beam-switching is normally realized by using phase shifters operating at RF or IF. A key to our proposed hardware-assisted initialization for the CMA adaptive antenna is additional hardware for estimating the received signal power from the beam. Consequently, the beam with maximum power will be exploited to initialize the CMA antenna. With reference to Fig. 3.4, the proposed system utilizes low-cost one-bit phase shifters for beam switching. In terms of implementation, the one-bit phase shifter can be easily realized by PIN diodes. Furthermore, power estimation can be achieved by a commercial power detector IC. Therefore, reliable initialization is possible yet with low-cost hardware. Finally, CMA adaptation can be performed by a digital signal processor (DSP) or FPGA. Note also that the technique is purely blind, *i.e.* it does not require the training sequence as opposed to the existing system presented in [21]-[22].

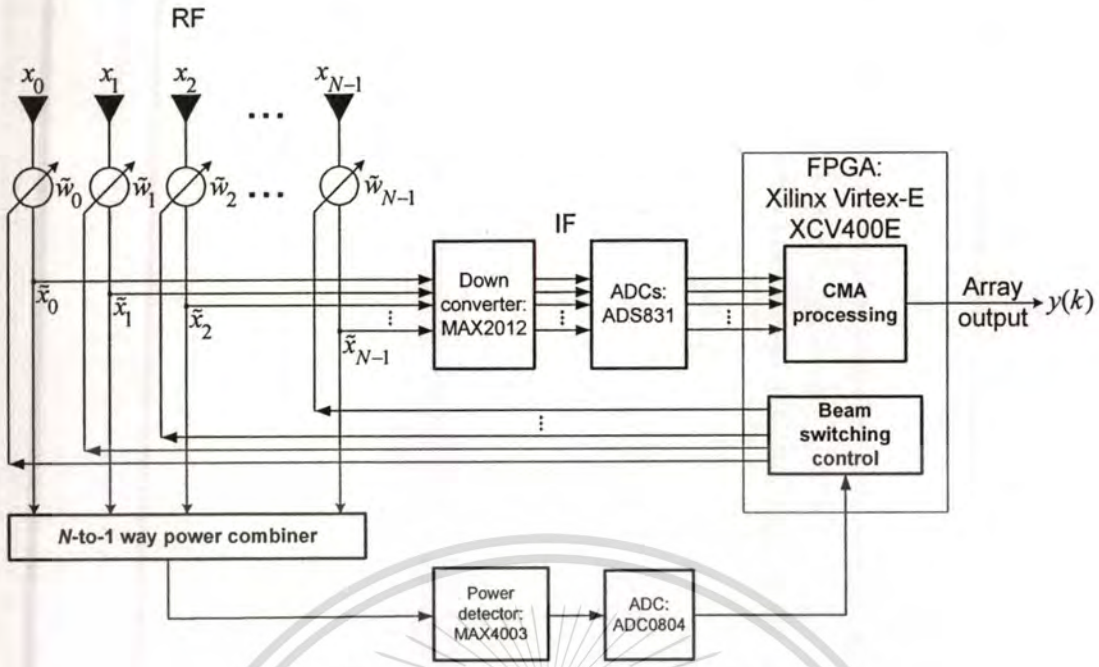


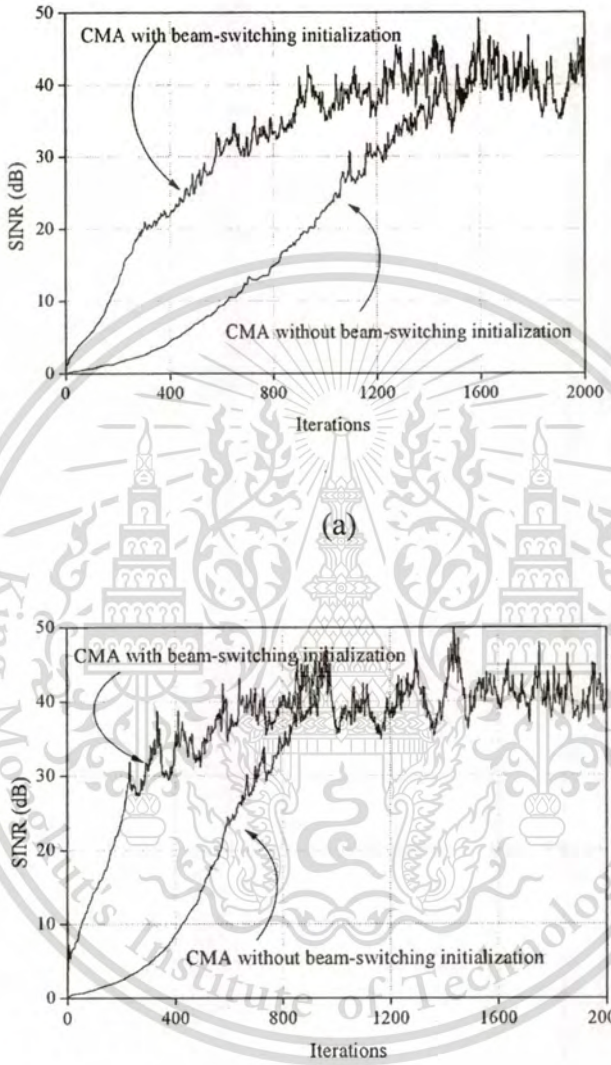
Fig. 3.4 CMA adaptive phased array architecture.

3.4 Simulations

3.4.1 Validation of the proposed concept

To validate the concept of hardware-assisted initialization for CM array, simulations were conducted. The proposed system was tested with a four-omni-directional-element circular array employed as a receiving antenna. Following the derivation in [37], the antenna can switch its main beam into four different directions, namely at ϕ_0 = 45°, 135°, 225°, 315°, by varying the phase-shift in accordance with Table 2.1. Although other beams can be formed using different phase-shift values, the underlying beams are adequate for this work. This is because the beamwidth of the omni-directional–element antenna is relatively wide i.e. >90°. Therefore, four beams can seamlessly cover all of the area, i.e. 360°. Nonetheless, the extension of the proposed technique for the narrow-beamwidth antenna is straightforward. Desired and interference signals were mutually independent π/4-QPSK and propagated through an additive white Gaussian noise (AWGN) channel. In practical wireless environments, the desired signal is normally larger than that of co-channel interference. Therefore, in our simulations, the power of the desired signal was set to be 3 dB stronger than that of the interference. Simulations were distinguished into two cases. In case 1, incident

angles of desired and interference signals were 30° and 80°, respectively. The simulation revealed that the main beam of our proposed antenna was initially in the direction of ϕ_0=45°, which is clearly an optimum beam among four possible beams realizable from the antenna.



(a)

(b)

Fig. 3.5 SINR trajectories (a) Case 1 and (b) Case 2.

Fig. 3.5 (a) depicts signal-to-interference plus noise ratio (SINR) trajectories of case 1 when employing the proposed technique and conventional CM array. The figure clearly shows that the convergence rate of the proposed adaptive antenna using hardware-assisted initialization is faster than that of conventional method [5].

Fig. 3.5 (b) depicts simulation results of case 2 where direction of the interference signal was further moved to ϕ_i = 120°. It is apparent that the convergence rate of CMA with hardware-assisted initialization was also faster than that of the standard CMA. In addition, case 2 achieves faster convergence rate than case 1 since the direction of interference signal is far from that of the initial main beam.

Besides the SINR trajectory, the behavior of weights obtained from performing CMA was determined. Fig. 3.6 and Fig. 3.7 illustrate the trajectories of magnitude of weights obtained from performing CMA of case 1 with and without beam-switching initialization, respectively. Fig. 3.8 and Fig. 3.9 illustrate the trajectories of phase of weights obtained from performing CMA of case 1 with and without beam-switching initialization, respectively. According to Fig. 3.6 to Fig. 3.9, it is clear that the adaptive antenna using CMA adjusts both of magnitude and phase before converging to its solutions. Moreover, it is clearly seen that after convergence the solution of the CMA with beam-switching initialization is different from that of CMA without beam-switching initialization. This agrees with the fact that the CMA has the several local minima and one global minimum and its solutions depend upon initialization.

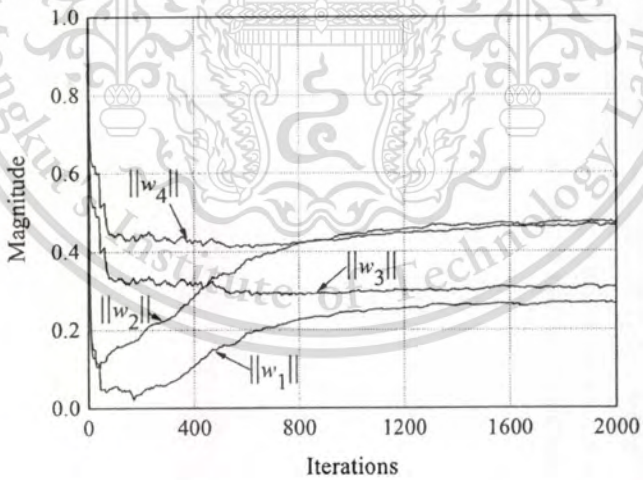


Fig. 3.6 Magnitude of weights in case1 of CMA with beam-switching initialization.

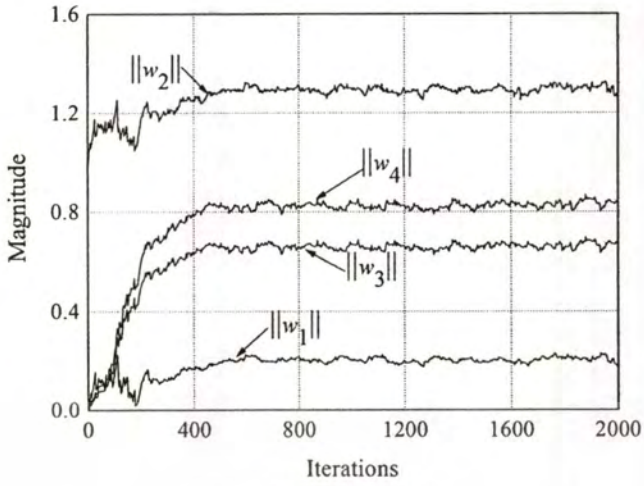


Fig. 3.7 Magnitude of weights in case1 of CMA without beam-switching initialization

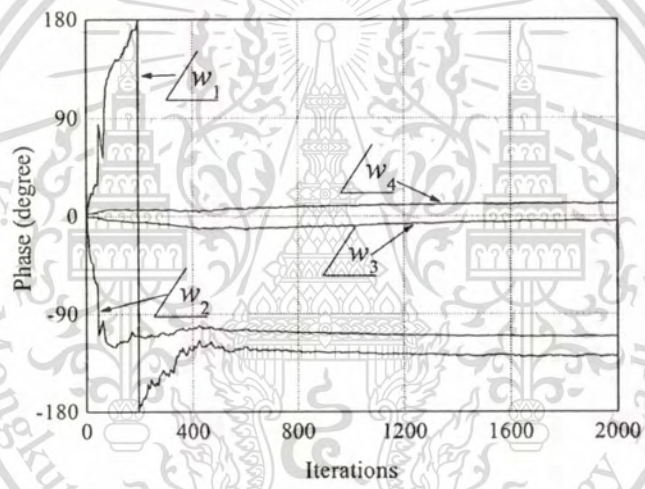


Fig. 3.8 Phase of weights in case1 of CMA with beam-switching initialization.

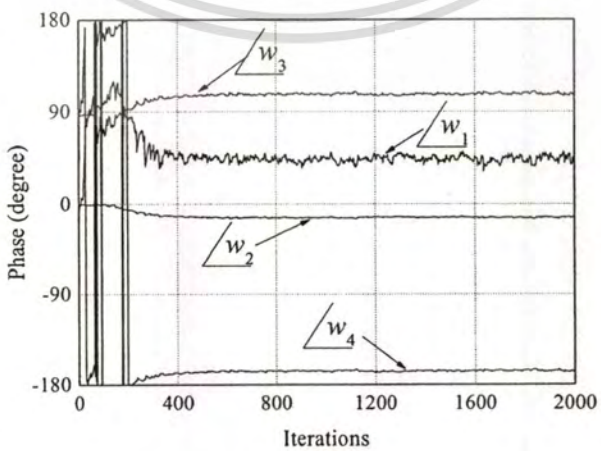


Fig. 3.9 Phase of weights in case1 of CMA without beam-switching initialization.

The effect of other interference directions on the proposed method is also examined. In the simulations, the direction of the desired signal is fixed at 30° while the incident angle of the interference signal is varied from 80° to 30°. Fig. 3.10 (a) and (b) respectively show SINRs obtained from the proposed adaptive antenna at 2,000 and 40,000 iterations against the angle space between the desired and interference signals. The results imply that the convergence rate of the proposed method depends on the angle space. Moreover, if the direction of the interference signal is identical to that of the desired signal, the CMA as same as other algorithms fail to recover the desired signal.

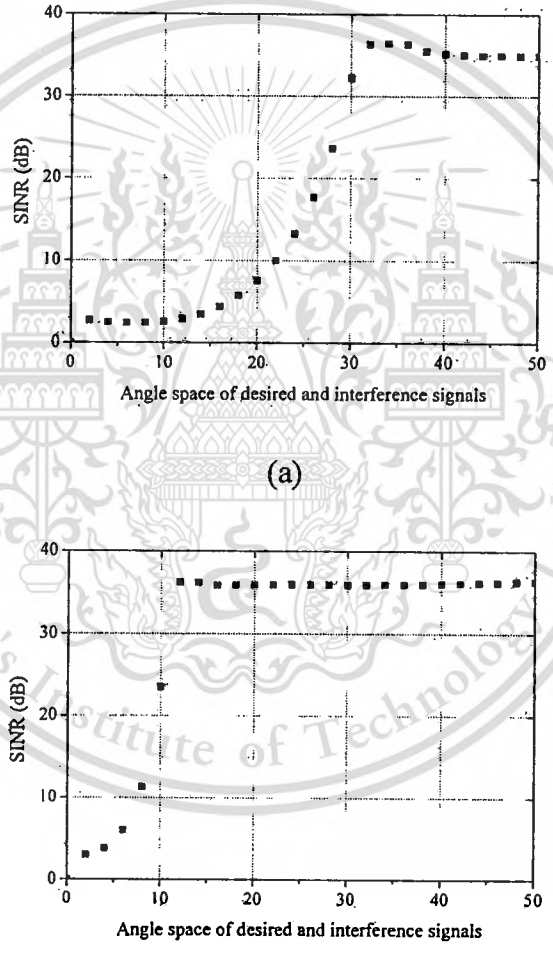


Fig. 3.10 SINR obtained from the proposed adaptive antenna at (a) 2,000 iterations (b) 40,000 iterations.

3.4.2 Additional case studies of the proposed adaptive antenna

So far, the additional case studies of the proposed antenna were presented with two cases. First, in simulations, there is an interference and desired signal. Second, the number of interference is increased to be two to investigate the convergence behavior of CMA.

3.4.2.1 One interference and one desired signals

In the first case, the convergence property of CMA was investigated to realize the initialization of CMA for the proposed adaptive antenna with the different initial main beam. To ensure that the total received power at the signal processing unit has an impact on the convergence property of CMA, the power of desired signal is set to be equal to that of interference one. Thus, the power of received signals relies on the antenna pattern and the direction of arrival. In the simulations, the desired and interference signals were $\pi/4$-QPSK modulated and propagated through an AWGN channel with 20 dB. The step size is set to be 2^{-9}. The simulations were distinguished into two cases. In case 1, incident angles of desired and interference signals were 80° and 30°, respectively. With the initial main beam being $\phi_0 = 45°$, this was conducted to show the convergence behavior of CMA when the interference signal is closer to the initial main beam direction than the desired signal. In this situation, the initial SINR is -0.9 dB. Fig. 3.11 illustrates SINR trajectories of CMA when employing the proposed adaptive antenna. The figure reveals that CMA has misconvergence. The antenna captures interference rather than the desired signal. Then, the initial main beam direction was changed to be $\phi_0 = 135°$ to resolve the problem of misconvergence. The initial SINR is 6.5 dB. According to Fig. 3.11, the SINR obtained from performing CMA is more than 0 dB at all. Fig. 3.12 illustrates the antenna pattern obtained from CMA after convergence.

Another matrix to determine the behavior of CMA is a signal constellation diagram. The signal constellations of output of CMA after 2,000 iterations with initial main beams being $\phi_0 = 45°$ and 135° are shown in Fig. 3.13 (a) and (b), respectively. The phase offsets of signals are about -51.3° and 9.6° for $\phi_0 = 45°$ and 135°, respectively. These results imply that the output signals obtained from CMA processing should be rotated or encoded before decision for the signals.

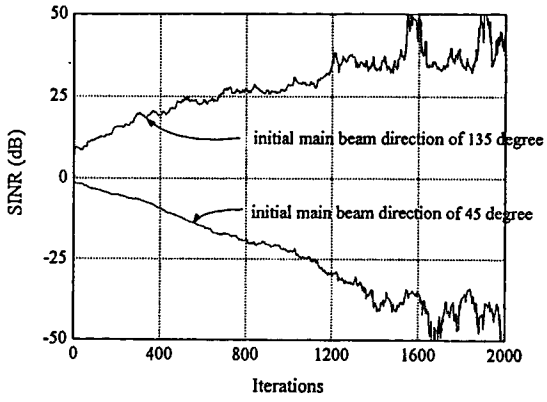


Fig. 3.11 SINR trajectories of CMA with two different initial beams.

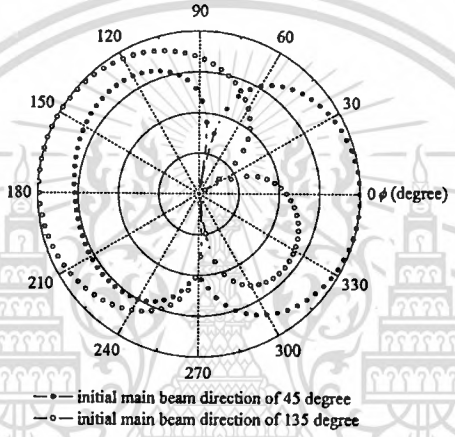
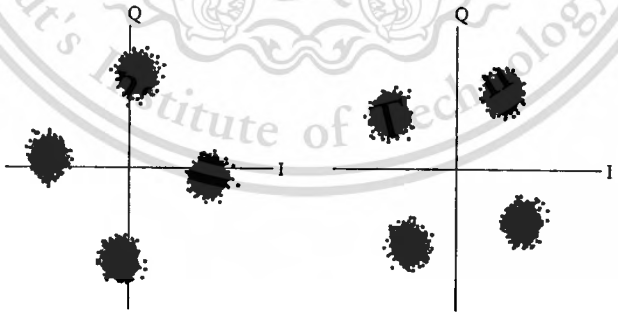


Fig. 3.12 Antenna pattern of CMA after convergence.



(a) (b)

Fig. 3.13 Signal constellation of CMA after 2,000 iterations with ϕ_0 = (a) 45° (b) 135°

For case 1, the incident angles of interference and desired signals are close to the forth and second quadrants, respectively. To guarantee that the antenna captures the desired signal, the beam whose main beam direction is closest to the desired signal

should be selected as the initial beam. In case 2, the investigation of CMA behaviour was conducted in the same way of case 1 but the incident angles of interference and desired signals were changed to be 10° and 55°, respectively. With ϕ_0 = 45° and 315°, the initial SINRs are -1.0 dB 5.6 dB, respectively. The SINR trajectories obtained from performing CMA were illustrated in Fig. 3.14. In the figure, the CMA adaptive antenna with the initial main beam of 45° captures interference while the antenna with the initial main beam of 315° can capture desired signal. Fig. 3.15 shows the final antenna pattern of the adaptive antenna using CMA after convergence. In this case, miscapturing of CMA can be resolved by switching the initial main beam to be 315°.

Fig. 3.16 shows the signal constellation diagram of output of CMA after 2,000 iterations. The signal constellation of output of CMA with the initial main beam being ϕ_0 = 45° and 315° respectively reveals the phase offset of -27.6° and 3.4°.

According to simulations of case 1 and 2, it is clear that the CMA will capture the desired signal if the initial SINR is positive. On the other hand, CMA will capture interference if the initial SINR is negative. Note that the initial beam for CMA is definitely necessary.

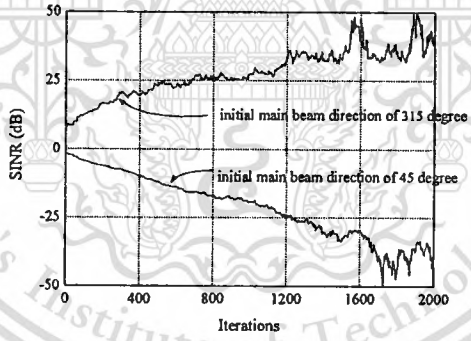


Fig. 3.14 SINR trajectories of CMA with two different initial beams.

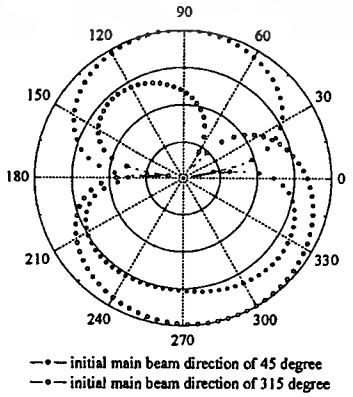
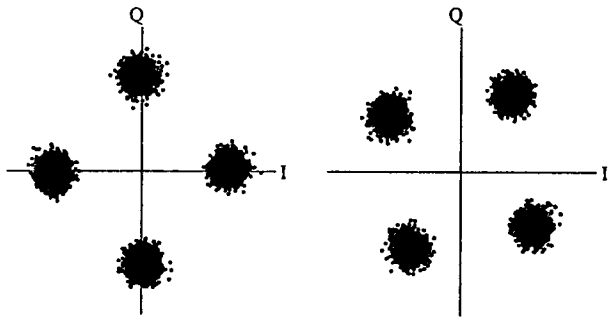


Fig. 3.15 Antenna pattern of CMA after convergence.



(a) (b)

Fig. 3.16 Signal constellation of CMA after 2,000 iterations with ϕ_0 = (a) 45° (b) 315°

3.4.2.2 Two interference and one desired signals

In this case, the convergence property of the proposed adaptive antenna was investigated via the simulation to realize that the proposed adaptive antenna can eliminate more than one interference signal. The simulation was conducted in the similar way of that with one interference signal. The incident angles of the desired and two interference signals were 30°, 80° and 120°, respectively. Fig. 3.17 illustrates the SINR trajectory obtained from the simulation. In the figure, the SINR is increased and reveals that the proposed adaptive antenna can eliminate two interference signals. The antenna pattern obtained from performing CMA after convergence (at 20,000 iterations) is shown in Fig. 3.18. In the figure, the nullity appears at 80° and 120° which agree with the directions of the interference signals.

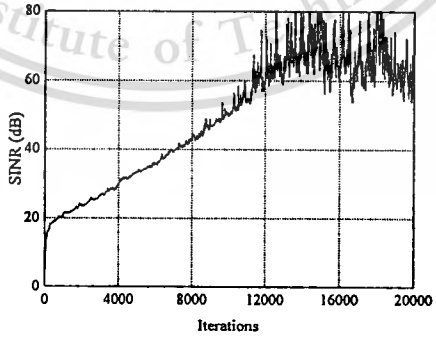


Fig. 3.17 SINR trajectory of CMA when two interference and one desired signals.

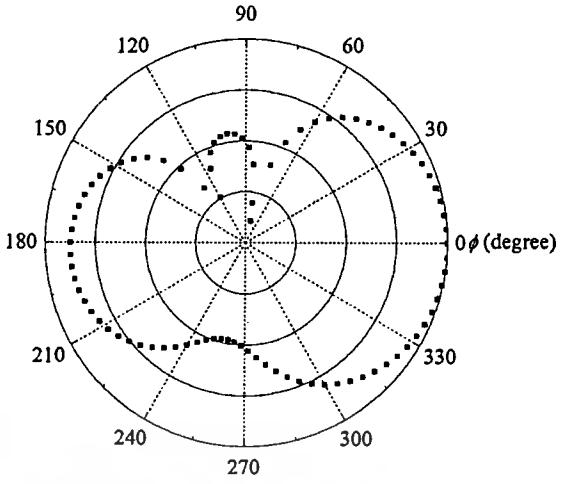


Fig. 3.18 Antenna pattern obtained from performing CMA with two interference signals at 20,000 iterations.

3.4.3 Effect of step size on the proposed adaptive antenna

The effect of step size on the proposed adaptive antenna was investigated via simulations. Simulations were conducted to guarantee that the proposed adaptive antenna should capture the desired signal (not interference). The desired and interference signals were π/4-QPSK modulated and propagated through an AWGN channel. The incident angles of desired and interference signals were 30° and 80°, respectively. The initial main beam direction was set to be 45° to carry out the optimal solution for CMA. The convergence behavior of CMA was considered when varying the step size and the level of noise.

Fig 3.19 illustrates the SINR trajectories of CMA with different values of step size μ when there is no noise. In the figure, when $\mu = 2^{-7}$, the proposed adaptive antenna has misconvergence. The antenna does not converge at all when μ is more than 2^{-7}. This means that the SINR trajectories in those cases are not stable. In contrast, the antenna can converge to capture the desired signal after decreasing μ to be less than 2^{-8}. The proposed adaptive antenna using CMA with $\mu = 2^{-8}$, 2^{-9} and 2^{-10} respectively converge at around 7,000, 13,000 and 17,000 iterations as shown in the figure. Subsequently, noise was added into the system under our consideration and its level in term of signal-to-noise ratio (SNR) was varied. Fig. 3.20 to Fig. 3.22 show SINR trajectories of the proposed antenna in case of the presence of noise. Comparing to the case of the absence of noise, the trajectories in case of the present of noise has

more fluctuations. As seen in the figures, when the SNR was decreased, the antenna with low SNR will converge to the optimal solution faster than that with high SNR. However, the SINR obtained from the proposed adaptive antenna using CMA after convergence was significantly decreased. For example, with $\mu = 2^{-8}$, although the antenna in case of the absence and presence of noise respectively converge at around 7,000 and 1,200 iterations, the resulting SINR are 297 dB and 49 dB as seen in Fig. 3.19 and 3.20, respectively.

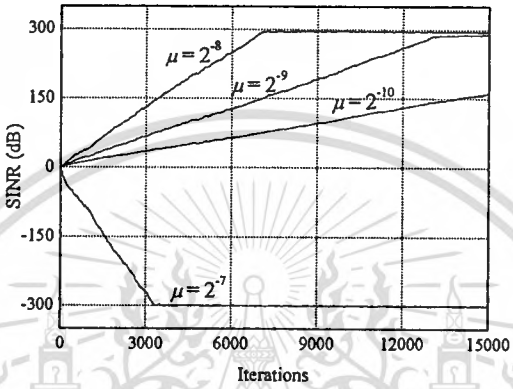


Fig. 3.19 SINR trajectories in case of the absence of noise.

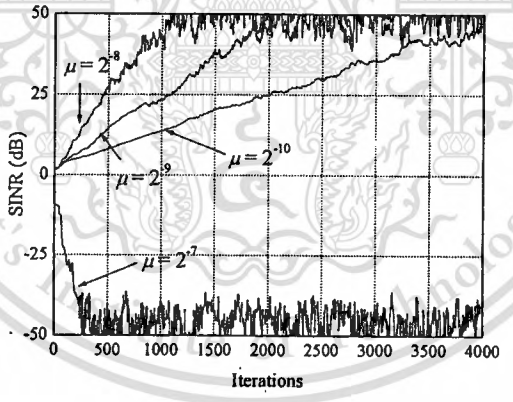


Fig. 3.20 SINR trajectories in case of the presence of noise (SNR = 30dB).

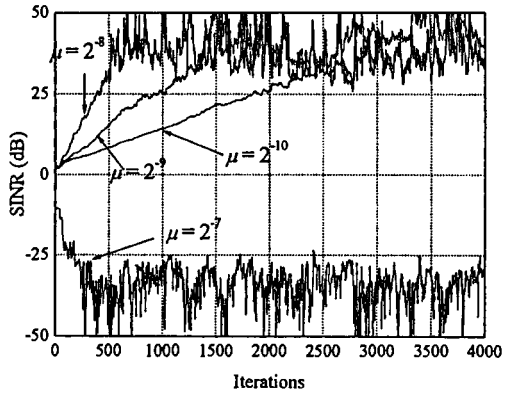


Fig. 3.21 SINR trajectories in case of the presence of noise (SNR = 20dB).

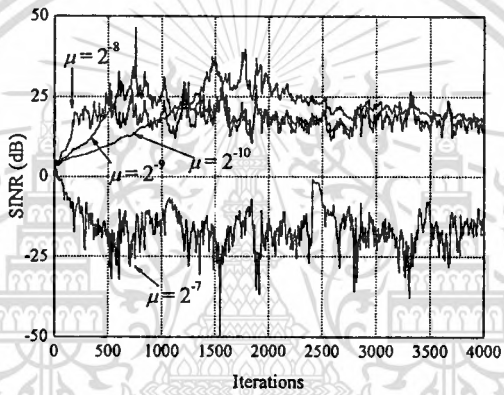


Fig. 3.22 SINR trajectories in case of the presence of noise (SNR = 10dB).

3.4.4 Multipath Fading Reduction

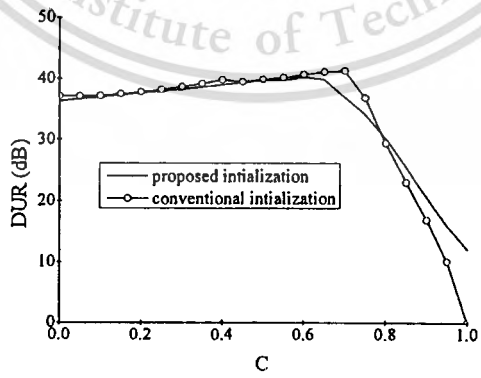


Fig. 3.23 Output DUR versus correlation coefficient.

In [50], the behavior of the LMS adaptive antenna in the presence of correlated signals was considered. They found that the ability of multipath fading reduction of LMS adaptive antenna relies upon the correlation coefficient between the desired ($d_k(t)$) and undesired ($u_k(t)$) signal. When the correlation coefficient is small, the antenna can suppress the undesired signal. Although the antenna fails to reduce the undesired signal when the correlation coefficient is close to one, the mean-square error is less than the thermal noise power. Thus, the signal fluctuation is negligible. In this section, we investigate the multipath fading effect on the proposed adaptive antenna. In our simulation, antenna parameters are almost identical to the previous section. The desired (30°) and undesired (120°) signals with the same power was π/4-QPSK-modulated and propagated through an AWGN (20 dB) channel. The power of the desired, undesired signals and thermal noise on each antenna element are represented by D_i, U_i and N_i, respectively. The multipath signals are in general correlated with one another. Let Ce^{-jb} be the correlation coefficient of the desired and undesired signals defined as

$$Ce^{-jb} = \frac{< d_1^*(t) u_1(t) >}{\sqrt{D_i U_i}} \tag{3.1}$$

where * and <> denote the complex conjugate and ensemble average. Here, b is set to be zero. Fig. 3.23 shows the desired-to-undesired signal power ratio (DUR) at the array output after CMA converges. The figure reveals that the ability of multipath fading reduction of the adaptive antenna depends upon the correlation coefficient between the desired and undesired signals. When C is close to 1, the CMA adaptive antenna with conventional initialization fails to reduce the undesired signal. However, the antenna with the proposed initialization (an initial main beam being $\phi_0 = 45°$) still can mitigate the multipath fading effect. This is because the initial power of the desire signal is stronger than that of the undesired signal. Despite the limitation of multipath fading reduction due to correlated signals, it does not have an impact on operation in an IMT2000 system since the transmitted signals must be spread by pseudo noise (PN) sequences. The correlation between two versions of the PN sequence delayed by one or more chips is almost zero. Thus, CMA adaptive antenna can be in fact applied for multipath fading reduction in IMT2000 communication system.

3.5 CMA Adaptive Processor

The processing unit of the proposed adaptive antenna using CMA was implemented on FPGA. The main operations in CMA processing include addition, subtraction and multiplication. In this section, the floating-point arithmetic was utilized to design the operators. The floating-point arithmetic represents the number in such a way that its binary-point is not fixed. Although hardware implementation of floating-point arithmetic is complicated, its accuracy makes itself attractive to CMA adaptive processor.

3.5.1 Floating-point representation

The floating-point arithmetic operation was designed based upon IEEE standard 754 with single precision. Floating-point numbers usually comprise a multiple of size of a word. The representation of a floating-point number is shown in Fig. 3.24, where s is the sign of the floating-point number (1 means negative), exponent is the value of the 8-bit exponent field (including the sign of the exponent), and significand (mantissa) is 23-bit number in the fraction. This representation is called sign and magnitude, since the sign has a separate bit from the rest of the number. In general, floating-point numbers are of the form [51]

$$(-1)^s \times (1+\text{significand}) \times 2^{(\text{Exponent-Bias})} \tag{3.2}$$

where exponent bias for single precision is 127.

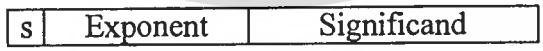


Fig. 3.24 Floating-point representation.

3.5.2 Floating-point addition and subtraction

Addition and subtraction of floating-point numbers can be divided into four consecutive parts: check for zero, align the significand, add or subtract the significand and normalize result. Fig. 3.25 and Fig. 3.26 respectively show a flow chart and block diagram of an arithmetic unit dedicated to floating-point addition and subtraction. A floating point number that is zero can not be normalized. If this number is used during the computation, the result may also be zero. Therefore, zero-check during the

normalization process is used to eliminate the problem. In the significand alignment process, the exponent of one operand is subtracted from the other using ALU to determine the larger one and the difference between each number. Then, a significand of the smaller number is right-shifted, and its exponent is increased accordingly. The significand is added or subtracted depending on the operation and the signs of the two significands. The exponent is increased or decreased and the significand is shifted left for normalization.

The adder and subtractor were designed to be combinational logic circuits along with a register as seen in Fig. 3.27. The register was placed after combinational logic circuit to reduce the critical path of the adder and subtractor. It is also used to synchronize to other operators.

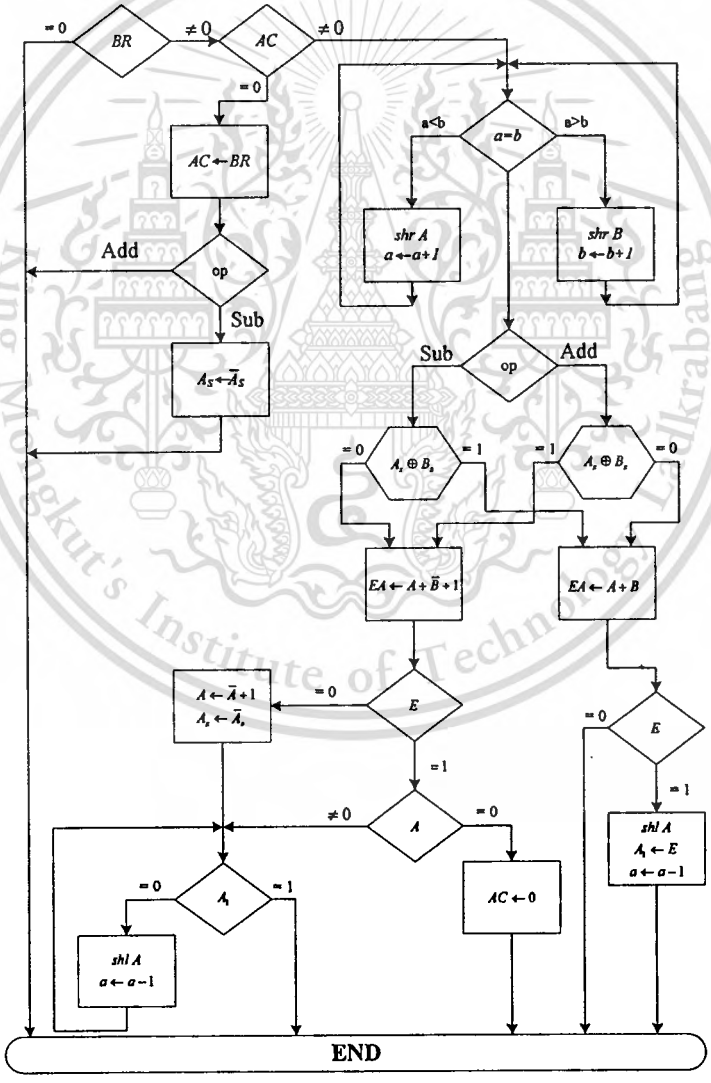


Fig. 3.25 Flow chart of floating-point addition and subtraction.

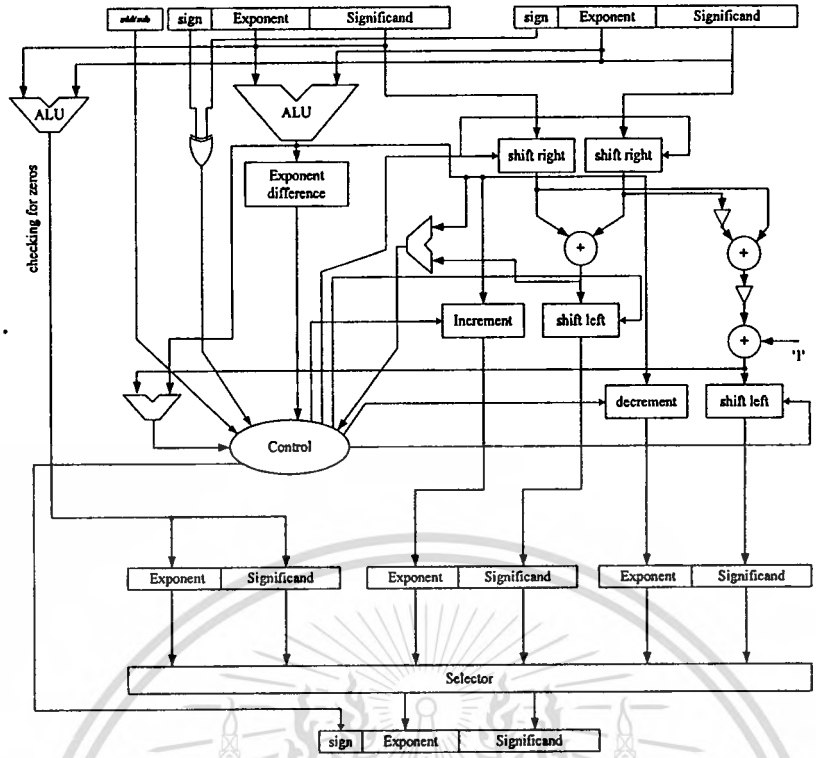


Fig. 3.26 Arithmetic of Floating-point addition and subtraction.

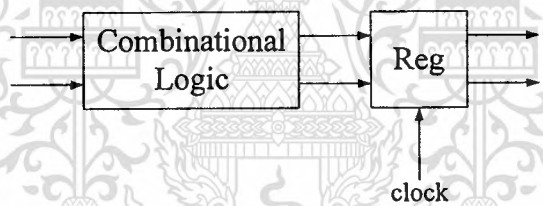


Fig. 3.27 Architecture of combinational logic circuit along with registers.

3.5.3 Floating-point multiplication

Multiplication of two floating-point numbers requires sinificand multiplication and exponent addition as shown in the flow chart of Fig. 3.28. As opposed to addition or subtraction, comparison of exponents or alignment of significand is not necessary. Multiplication of floating-point numbers can be subdivided into four parts: check for zero, add the exponents, multiply the significand and normalize the product. If either operand is equal to zero, the product is zero. Both exponents of two numbers are added. The exponent sum will have double bias. The correct biased exponent for the product is obtained by subtracting the bias number from the sum. The multiplication of the significand is performed as similar to the fixed-point case. The block diagram of an arithmetic unit dedicated to floating-point multiplication is shown in Fig. 3.29.

Similar to the adder and subtractor, the multiplier was designed and implemented to be combinational logic circuits along with a register.

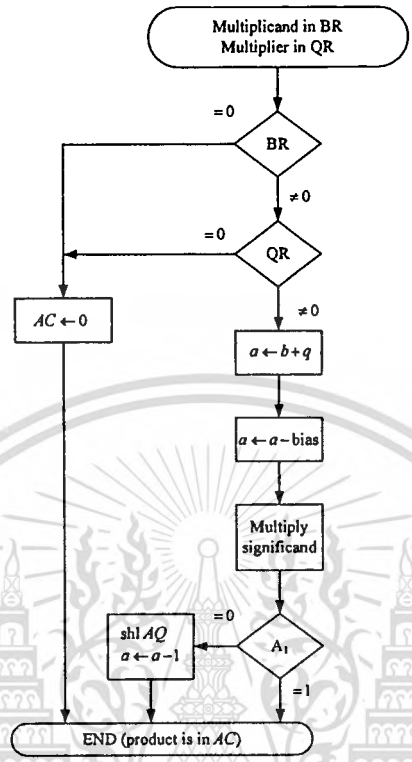


Fig. 3.28 Flow chart of floating-point multiplication.

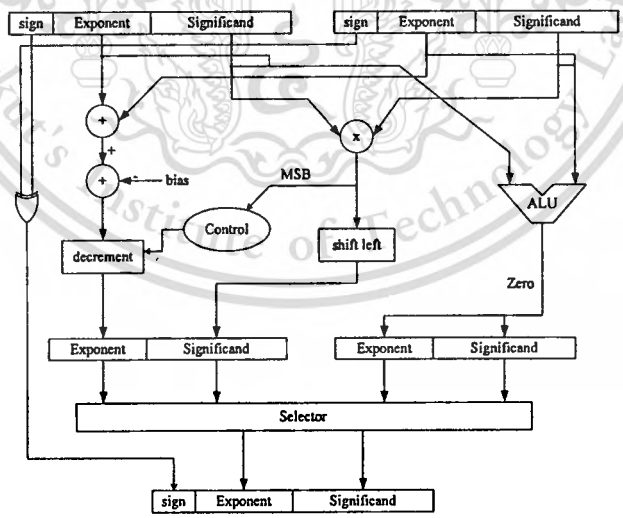


Fig. 3.29 Arithmetic of Floating-point multiplication.

3.5.4 Floating-point conversion

Conversion between fixed-point and floating-point numbers is required, as most commercial data converters are fixed-point devices. Fig. 3.30 shows a block

diagram of conversion from a signed integer (fixed-point) number to the corresponding floating-point number. Fig. 3.31 shows the method of conversion from floating-point output of the CMA processor to signed integer number for D/A.

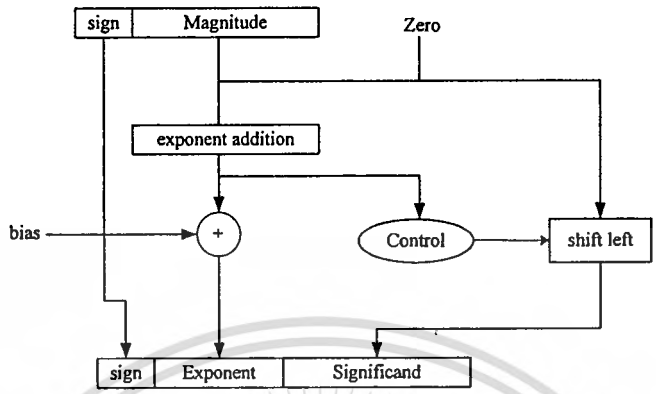


Fig. 3.30 Arithmetic of conversion fix-point number to floating-point number.

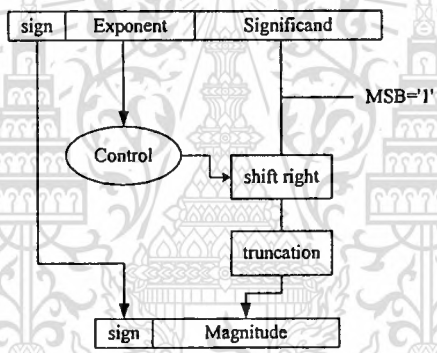


Fig. 3.31 Arithmetic of conversion floating-point number to fix-point number.

3.5.5 CMA processor based on MAC operation

Fig. 3.32 depicts the structure of a signal processing unit for adaptation of CMA according to Eq. (2.18). Essentially, it includes a floating-point "multiply and accumulate" (MAC) processor together with multiplexers, demultiplexers and latch. Although hardware implementation of floating-point arithmetic is complicated, its accuracy makes itself attractive to CMA adaptive processor realization. More specifically, floating-point operation can reduce problems of truncation and computation of small step-size μ governing convergence rate of CMA. Control circuitry is required since the CMA processor must be reused for each input sample received from each antenna element.

Referring to Fig. 3.32, two 8-channel multiplexers are used to select each i^{th} element of the vector $\tilde{\mathbf{x}}(k)$ and $\tilde{\mathbf{w}}(k)$, i.e. $\tilde{x}_i(k)$ and $\tilde{w}_i(k)$ required for computation of

Eq. (2.18). Eight channels are required because the element of each vector is complex. One multiplexer output is fed to the number conversion unit to convert $\tilde{x_i}(k)$ from a fixed-point to a floating-point format. Two 4-channel multiplexers are used to select signals for the MAC processors. Demultiplexing is necessary after the MAC operation for decomposing the real and imaginary parts of $y(k)$. An additional 8-channel demultiplexer is employed for monitoring the weight vector $\mathbf{w}(k)$. The processor operates in real-time at 16 MHz master clock. The FPGA implementation of 29,878 gates consumes 1,504 slices, parts of which are 2,078 look-up tables (LUTs).

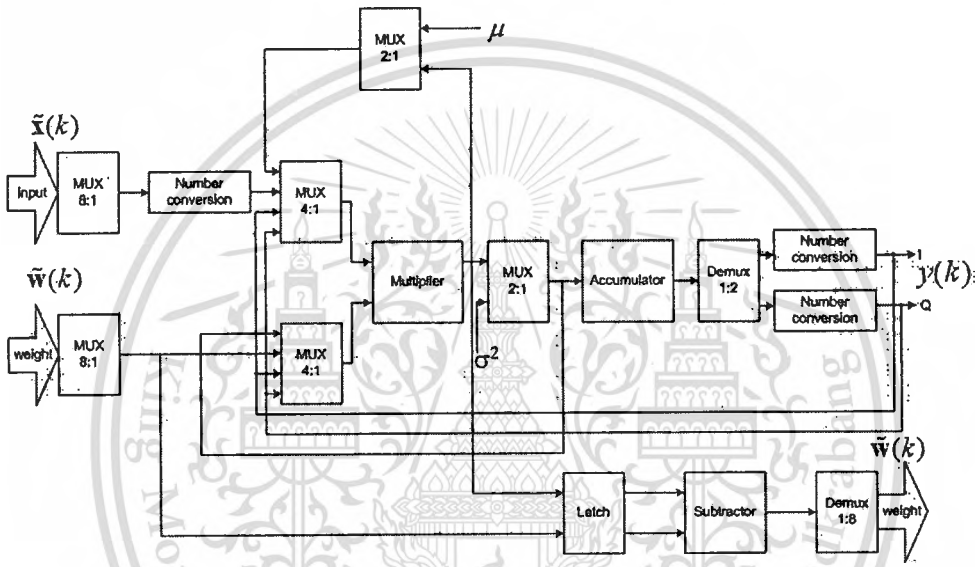


Fig. 3.32 CMA adaptive processor architecture.

3.6 Antenna Prototype

Based upon the concept described above, a hardware prototype was realized as shown in Fig. 3.33 and its specification is summarized in Table 3.1. In general, the system comprises RF chain, power detector, analog to digital converters (ADCs), and signal processing unit. Basically, it takes the RF signal from each element of a received antenna. Then the RF signal is converted from 1.95 GHz down to IF signals of 50 kHz through four I and Q down converters (MAX2102) and analog filters. Each I and Q IF signal is amplified by the OPA842 op-amp, digitized by ADS831 ADCs at a sampling rate of 250 kHz and is fed to the signal processing unit, which effectively operates at IF.

The associated control circuitry and adaptive signal processor are implemented on Xilinx Virtex-E XCV400E FPGA evaluation kit. Note that interfacing between the evaluation kit and PC is achieved via parallel port. Several control-functions are performed on the FPGA. Firstly, it generates control signals for one-bit phase shifters, i.e. PIN diodes used for beam-switching. Next, it compares the RF received signal power of each individual beam measured via a power detector MAX4003. The beam with maximum power will be selected as the initial beam for CMA as described previously. Note that the ADC sampling clock is synchronized with the clock used in the adaptive processor unit.

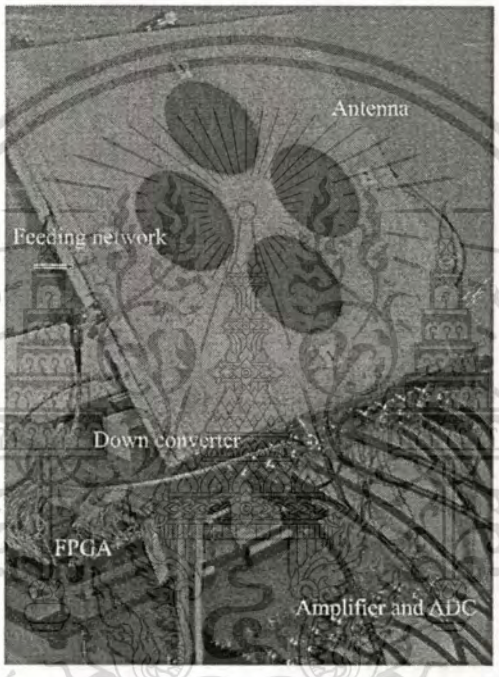


Fig. 3.33 Proposed adaptive antenna prototype.

Table 3.1: Prototype specifications

Receiving antenna	A flat four-beam compact phased array antenna
RF frequency	1.95 GHz
IF frequency	50 kHz
Modulation	π/4-QPSK
Modulation bit rate	10 kbps
ADC sampling rate	250 kHz
ADC	Eight 8-bit channels
ASIC for DSP	Xilinx Virtex-E XCV400E FPGA
Master clock frequency	16 MHz

3.7 Experimental Results

Fig. 3.33 illustrates a prototype of the proposed adaptive antenna system. Before evaluating performance of the whole system, the phased array antenna was tested. It was found that VSWRs of four input and one output ports at 1.95 GHz measured by HP8510C network analyzer were 1.26, 1.17, 1.24, 1.18 and 1.45, respectively. Furthermore, VSWR for all ports in a designed frequency band of 1.92-1.98 GHz was in the range of 1.5:1.

Next the adaptive antenna system using the proposed initialization was examined in an anechoic chamber. Direction of the desired and interference signal was 30° and 120°, respectively. Both signals were $\pi/4$-QPSK-modulated with mutually independent 10 kbps data. The distance between the receiving antenna and transmitting antennas was 2.2 m and height of both antennas was 0.5 m. A transmitted signal power was adjusted such that SNR at the output of the receiving antenna was 20 dB and that the desired and interference signal power was 3 dB different. The circular patch element was also used for both desired and interference transmitted antennas. Experimental results were compared with the simulation results. In the simulation, the closed-form approximation of the circular-patch parameters based on a cavity model [39], [52] is used. That is, the patch was modeled as two electric conductors at the top and bottom representing the patch and a ground plane, respectively.

Fig. 3.34 depicts the measured initial antenna pattern along with the simulated pattern. The pattern was measured right after automatic beam-scanning was achieved. At the clock frequency of 16MHz, the total time used for determining the initial beam is well below 500ns. It is seen that the direction of the main beam with maximum received power ϕ_0 was 60°. The difference from expected ϕ_0 of 45° is mainly due to effect of the circular-patch element. At any rate, four different beams still covered all angles. The beam was automatically employed as the initial beam, which achieved SINR of 3.6 dB. Then, CMA beam-forming was performed. Fig. 3.35 shows the measured antenna pattern along with the simulated pattern after only 1,100 iterations which is equivalent to 4.4 μs at the sampling frequency of 250 kHz. It is seen that the main beam was in the direction of 40° and the nullity appeared at 123°. This is respectively in a good agreement with the direction of the desired (30°) and

interference signal (120°). Although the experimental results do not exactly resemble the simulation ones, the antenna is still able to nullify the interference signal. The difference in the results obtained from simulation and experimentation is mainly due to the mutual coupling effect among the array elements and the diffraction caused by the conducting plate [53]. More importantly, SINR achieved from the proposed system at this instant became 22.1 dB as compared to the initial value of 3.6 dB. Note that the SINR obtained from the simulation was 33.2 dB.

Fig. 3.36 illustrates a signal constellation attained from the output of the proposed system (Fig. 3.36 (a)), which indicates that CMA successfully converged as the IF signal envelope become constant. Dispersion in signal constellation was due to non-idealities in the down converter, e.g. frequency drift in local oscillator and additive noise. Next the IF signals were down-converted to baseband by means of computer software using an ideal mixer and a low-pass filter. The baseband output constellation is shown in Fig. 3.36 (b), which clearly reveals a QPSK constellation. Although further performance improvement in RF circuitry is necessary for perfect operation of the antenna system, the results discussed in this section confirm the feasibility of the proposed low-cost hardware-assisted initialization technique.

Finally, when the direction of the interference signal was changed to 80°, it was observed that the initial SINR is 0.3 dB. After performing CMA, the measured pattern along with simulated one is shown in Fig. 3.37. The SINR of the proposed system converges to 19.3 dB. As expected from the simulations above, the convergence rate in this case is slower i.e. after 1,800 iterations. This is because the initial SINR of the latter case is less than that of the former case [53].

3.5 Summary

In this chapter, a low-cost approach for initialization of a constant modulus array has been proposed. The technique relies upon simple hardware circuitry, i.e. one-bit phase shifters and a power detector. A hardware prototype utilizing FPGA and associated RF devices has been developed to validate the proposed concept. Simulations have shown significant improvement in convergence properties obtained from the proposed initialization technique. Moreover, the proposed system has been tested with the flat four-beam compact phased array antenna designed for operation at

IMT2000 frequency of 1.95 GHz. Experimental results have confirmed the feasibility of the proposed low-cost hardware-assisted initialization technique.

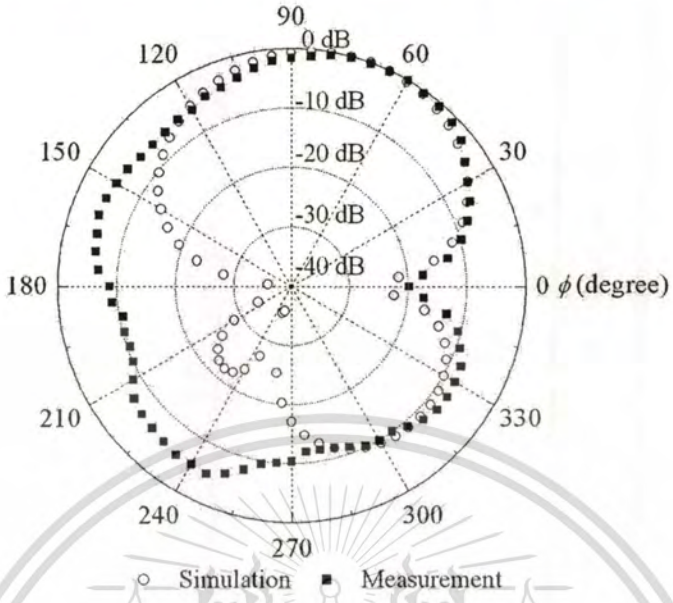


Fig. 3.34 Initial patterns for CMA.

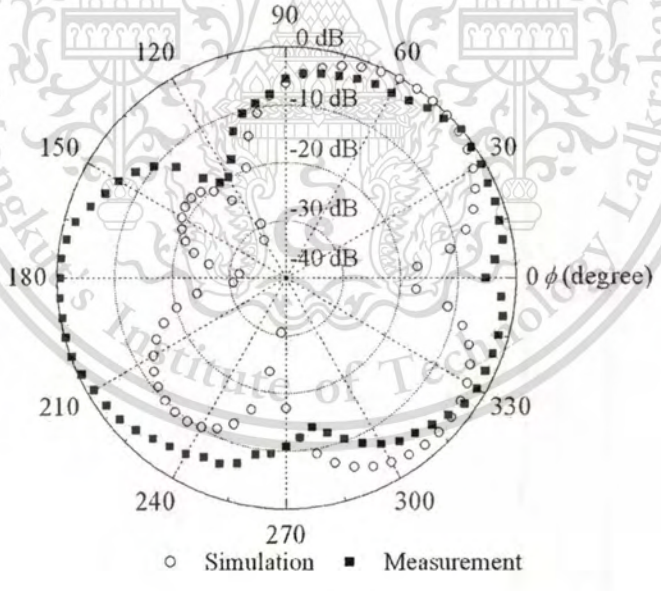
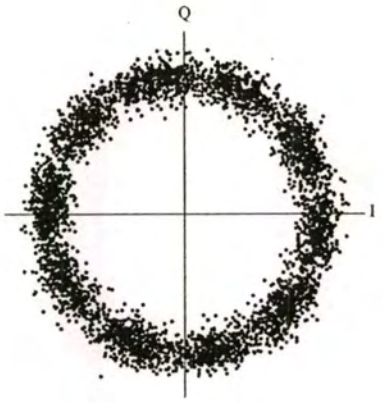
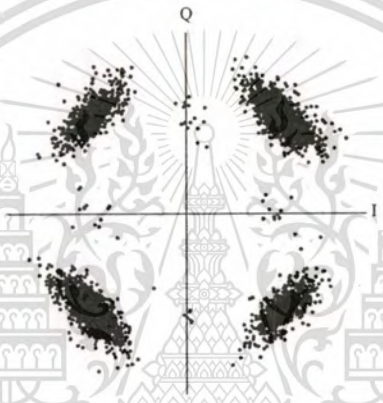


Fig. 3.35 Final patterns obtained from CMA after convergence (1,100 iterations).



(a)



(b)

Fig. 3.36 Output constellation of the proposed antenna system after convergence at (a) IF and (b) baseband.

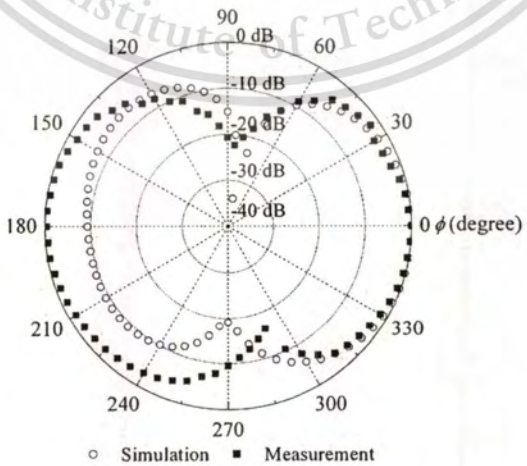


Fig. 3.37 Final patterns obtained from CMA after convergence (1,800 iterations).

CHAPTER 4
SWITCHED-BEAM ELEMENT PHASED ARRAY ANTENNA FOR CMA INITIALIZATION

4.1 Introduction

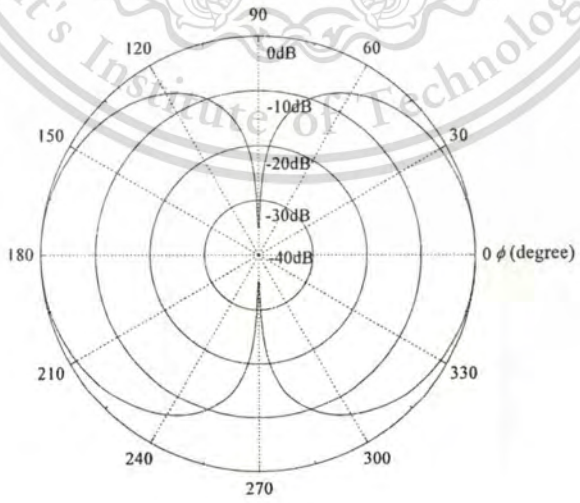
As proposed in Chapter 3, switching the main beam of an antenna to the direction close to a desired signal can improve the convergence rate of CMA. In this chapter, the beam switched into more different directions through phase shifters and applying forward or reverse bias to pin diodes is employed to initial CMA. The results will be shown that an array antenna which provides the narrow antenna pattern can improve the convergence rate of CMA also. Besides, the antenna which generates more directions of the switched narrow beams can resolve a problem that the direction of a desired signal is close to that of an interference signal.

4.2 Switched-beam Element Phased Array Antenna

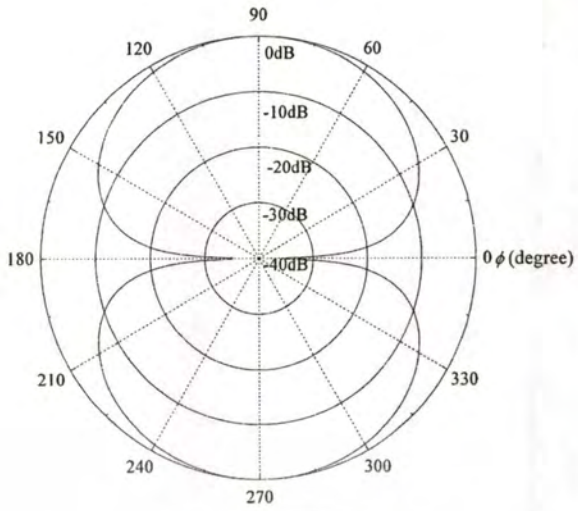
In [54], a switched-beam element was proposed. The antenna was designed to provide radiation patterns which can be switched into x and y directions in azimuth plane by applying forward or reverse bias to pin diodes, respectively. The antenna possesses bi-directional patterns as seen in Fig. 4.1. Subsequently, the switched-beam element was arrayed to provide signal-to-interference ratio improvement [55]. The design is based upon circular array principle.

In this chapter, the antenna discussed above is utilized in our adaptive antenna system. The peak beam direction θ_0 of 45° was defined to tilt the beam of the antenna for the specific coverage area. The configuration of the proposed adaptive antenna system is shown in Fig. 4.2. The antenna is a switched-beam element phased array which is based upon a circular array principle. Slight modification is made, particularly the array radius "r_a" of the phased array antenna was chosen as 0.3λ rather than 0.5 λ as originally presented in [55]. This is because we wish to obtain a desirable level of Front to Back ratio (F/B ≥ 10 dB). With one-bit phase shifters of ±54°, switching of the main beam into four directions, particularly at ϕ_0=45°, 135°,

225°, and 315° is possible. Selection of phase excitation for each antenna element was identical to what was described in [54]. The excitation was initially employed for switching of the main beam into four different quadrants. The number of switched-beams of the phased array antenna can be increased by applying forward or reverse bias for PIN diodes of the antenna elements. This is done because we wish that the angle-space between incoming signals and the initial main beam direction is possibly near. The new antenna can generate more antenna patterns by means of little additional hardware; it can mitigate the effect of beamwidth encountered in the previously proposed technique. Twelve beams covering the service area of 360° were selected for CMA initialization [56]. Then, the main beam is recursively adjusted in accordance with CMA. Table 4.1 summarizes the possible initial beams for CMA. Twelve beams which can cover the service area (360°) are selected and exploited to initial CMA. The control signal of the phase excitation for four one-bit phase shifter can be generated by the part of signal processing. A received power of each individual beam obtained from beam-switching is measured. This received power estimation can be accomplished by power detector. The decision to the maximum received power is performed in signal processing part. A beam with maximum received power will be selected as an initial beam for CMA. Note that the proposed adaptive antenna system is purely blind and easy to be further implemented with additional simple hardware *i.e.* power detector and analog to digital converter (ADC).



(a)



(b)

Fig. 4.1 Patterns of a switched-beam element in (a) x (b) y directions.

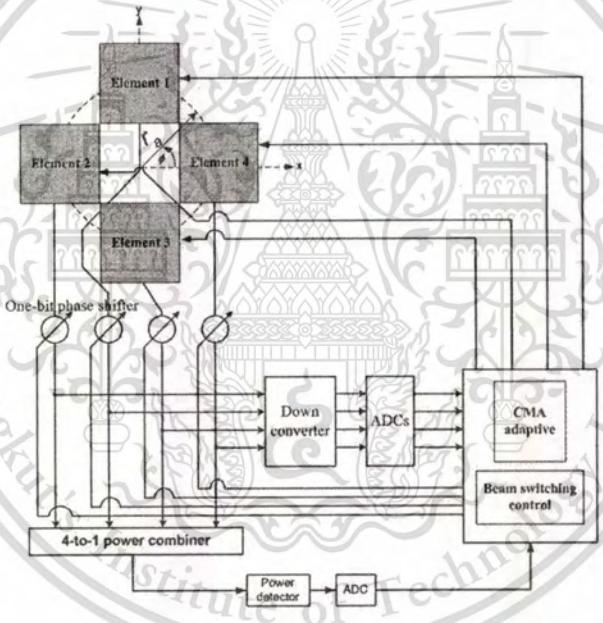


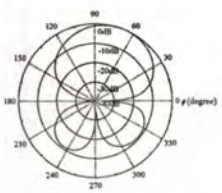
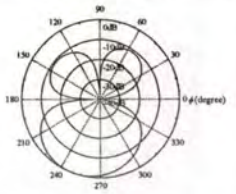
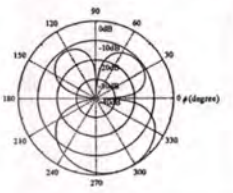
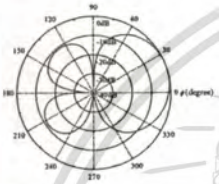
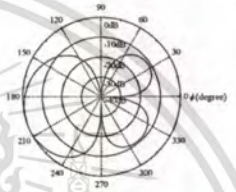
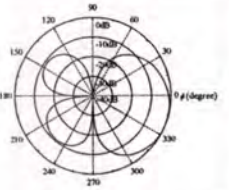
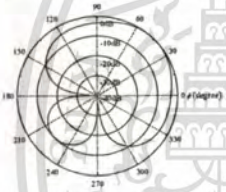
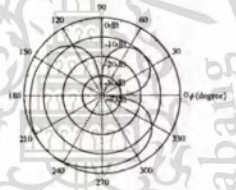
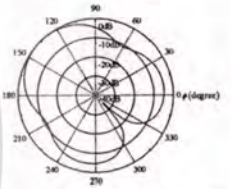
Fig. 4.2 Configuration of the proposed adaptive antenna.

4.3 Simulations

Simulations were conducted to evaluate the performance of the proposed adaptive antenna. The desired and interference signals were π/4-QPSK-modulated and propagated through an additive white Gaussian noise (AWGN) channel. In practical wireless environment, the received power of interference signal due to co-channel and multi-path fading is normally smaller than that of desired signal. Therefore, in our simulations the power of the desired signal was set to be 3 dB stronger than that of the

interference. Furthermore, parameters of the switched-beam element phased array antenna presented in [54] were used. SNR was set to be 20 dB.

Table 4.1 Possible initial beams of the switched-beam element phased array antenna

Element pattern (direction)				Total pattern by choosing phase shift values			
#1	#2	#3	#4	-54°, 54°, 54°, and -54°	-54°, -54°, 54°, and 54°	54°, -54°, -54°, and 54°	54°, 54°, -54°, and -54°
y	y	y	y				
x	x	x	x				
y	y	x	x				

Simulations can be distinguished into four cases. Firstly, the simulation was set to show that the proposed adaptive antenna system with the switched-beam element phased array antenna has a faster convergence rate than that with the flat four-beam phased array antenna. Secondly, the incident angles of desired and interference signals were set to verify that the convergence property of CMA with beam-switching initialization (switched-beam element phased array antenna) is better that of CMA without it. Thirdly, various beams obtained from beam-switching of the switched-beam element phased array antenna were employed to initial CMA. Fourthly, there are one desired and two interference signals in the simulation. This was conducted because we ensure that the proposed adaptive antenna can operate in the practical environment under our consideration.

In the first case, the angles of the desired and interference signals received by the switched-beam element phased array antenna were 80° and 50°, respectively. The direction of the initial main beam corresponding to maximum received power was ϕ_0 = 70°. This was attained by choosing phase shift of four elements to be -54°, 54°, 54°,

and -54° and by applying reverse bias to all antenna elements. SINR attained from the switched-beam element phased array antenna converges as soon as after 1,000 iterations. With the identical angle space between desired and interference signals, the angles of the desired and interference signals received by the flat four-beam compact phased array antenna were 35° and 65°, respectively. The direction of the selected initial main beam corresponding to maximum received power was $\phi_0 = 45°$. This was attained by choosing phase shift of the four elements to be -47.7°, 47.7°, 47.7°, and -47.7°. Fig. 4.3 shows that SINR obtained from performing CMA with the flat four-beam compact phased array antenna slowly increases and converges after 4,500 iterations while SINR obtained from performing CMA with the switched-beam element phased array antenna relatively fast increases and converges after 1,850 iterations. Both simulation conditions confirm that the convergence rate of CMA with the switched-beam element phased array antenna is faster than that with the flat four-beam compact phased array antenna. Note that angle-spaces of the desired and interference signals with respect to the initial main beam for both antennas are set to be identical since it has an impact on the convergence property of CMA as described in Chapter 3.

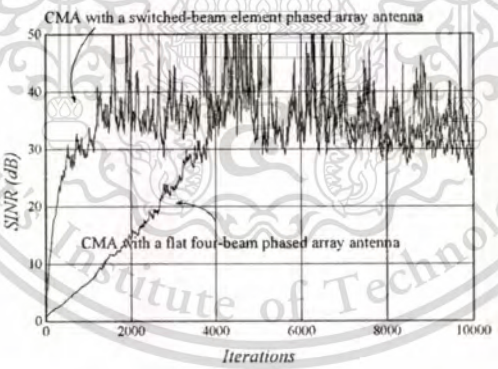


Fig. 4.3 SINR trajectories of CMA along with the flat four-beam compact phased array antenna and the switched-beam element phased array antenna.

The SINR trajectories of CMA from the second case are shown in Fig. 4.4 in which the incident angles of desired and interference signals were 70° and 120°, respectively. Here, it was assumed that a beam with maximum received power could be perfectly determined and was used for CMA initialization. With maximum received power, the main beam achieved by one-bit phase shifters was switched into the first quadrant from all of four quadrants. This was attained by choosing phase shift of four elements to be -54°, 54°, 54°, and -54°. Next, each antenna element was

applied to provide its radiation pattern being bi-directional in y direction. Therefore, the main beam employed to initial CMA directs to $\phi_0 = 70°$. Clearly, the proposed adaptive antenna (with beam-switching initialization) has faster convergence rate than the antenna without beam-switching initialization.

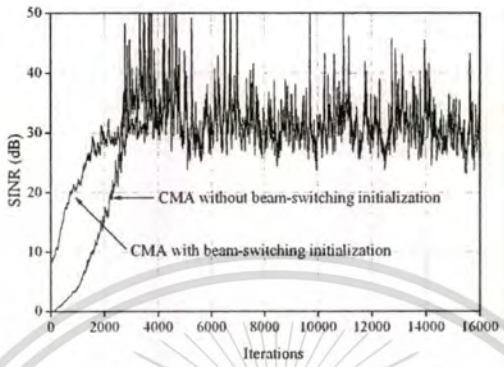


Fig. 4.4 SINR trajectories of CMA with and without switched-beam initialization.

With the same incident angles of signals, various beams obtained from beam-switching of the antenna were employed to initial CMA in the third case. The SINR trajectories of CMA with the initial main beam directions being $\phi_0 = 30°$, 45°, 70°, 110, 135, and 160 are shown in Fig. 4.5. Fig. 4.5 (a) reveals that the antenna with all initial conditions can converge by capturing desired signal rather than interference signal. In the figure, convergence rate of CMA with the initial main beam direction being $\phi_0 = 70°$ is fastest. In contrast, CMA with the initial main beam direction ϕ_0 of 30° has slowest convergence rate. These imply that convergence property of CMA relies upon not only an appropriate weight value [20] but also an initial beam. According to the figure, we can conclude that the convergence rate is increased if the direction of an initial main beam is far from interference signal.

Fig. 4.5 (b) shows the SINR trajectories when the direction of an initial main beam for CMA is in the second quadrant. Choosing phase excitation value and status of PIN diodes is in accordance with Table 4.1. In order to obtain the direction of the main beams being $\phi_0 = 110°$, 135° and 160°, phase shift values of four element were chosen to be -54°, -54°, 54°, and 54°. Besides, the antenna element (1 2 3 4) is bi-directional in x x x x, x x y y and y y y y directions for $\phi_0 = 110°$, 135° and 160°, respectively. The SINR trajectories of CMA in Fig. 4.5 (b) show that CMA converges to a wrong solution. Thus, the proposed adaptive antenna fails to retrieve the desired

signal. It captures the interference signal (not desired signal). This is because the initial main beam of the antenna is closer to the interference signal than the desired signal. This problem will be encountered if an initial beam is inappropriately defined. However, with reference to Fig. 4.2, scanning a main beam into all possible directions to find the maximum received power can resolve the underlying problem.

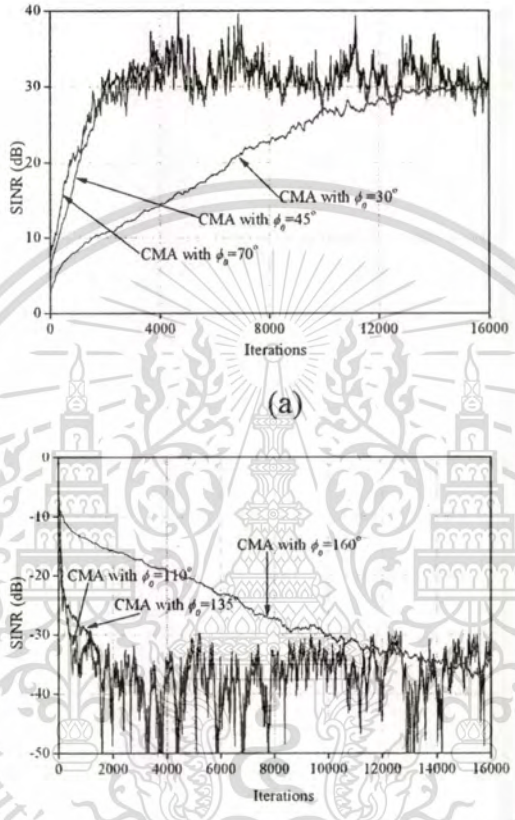


Fig. 4.5 SINR trajectories of CMA with various initial main beams in (a) first quadrant (b) second quadrant.

Finally, we wish to ensure that the proposed adaptive antenna can operate in practical environment under our consideration. Therefore, the number of interference signal is increased to be two. In the simulation, the angles of one desired and two interference signals were 45°, 95°, and 355° respectively. The direction of an initial main beam corresponding to maximum received power was ϕ_0=45. Fig. 4.6 shows the SINR trajectory of CMA in this case. The figure reveals that the antenna can mitigate the effect of two interference signals.

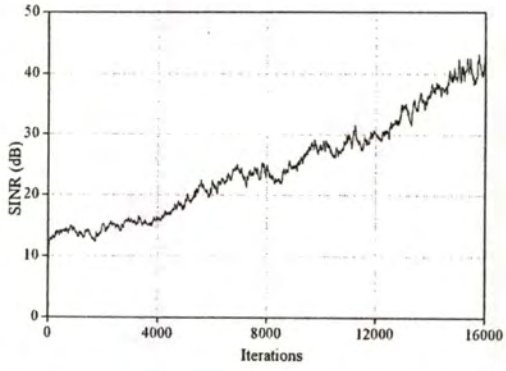


Fig. 4.6 SINR trajectory of CMA when there are one desired and two interference signals.

4.4 Summary

In this chapter, a technique to improve the convergence property of the CMA adaptive antenna system proposed in Chapter 3 by replacing the flat four-beam phased array antenna by the switched-beam element phased array antenna has been presented. The new antenna generate more pattern directions namely in twelve different directions covering the service area of 360°. The new antenna can resolve the problem of ambiguity in incident angles of desired and interference signals. The simulation results was validated that the proposed adaptive antenna system with the switched-beam element phased array antenna has the fastest convergence rate.

CHAPTER 5
DESIGN OF EFFICIENT ARCHITECTURE FOR CMA ARRAY PROCESSING UNIT

5.1 Introduction

In this chapter, an efficient architecture of the processing unit for the adaptive array antenna using CMA is presented. In order to achieve the efficient architecture which can operate at the high throughput, the pipelining technique is employed to reduce the critical path of the FPGA for implementation. With applying pipelining, the original CMA is modified by inserting the delays D into the feed-forward and error-forward paths. This modified algorithm can be referred to as the delayed CMA (DCMA). Before presenting the architecture, the DCMA is introduced along with its derivation. The convergence behavior of the original CMA and DCMA is considered and compared together via simulations. The simulation is divided into 2 cases, i.e. cases of a 2-element linear array antenna and the proposed adaptive antenna. The goal of the first case is to investigate the effect of inserting delays into CMA on the linear array antenna which is the most popular antenna used in adaptive antenna systems. The second case studies the behavior of DCMA along with the antenna proposed in Chapter 3. After that, the DCMA-based pipelined parallel architecture is presented to achieve the desirable critical path for hardware implementation. The fixed-point arithmetic is in general exploited to implement the adaptive processing unit in order to reduce the cost and power consumption. Thus, the finite worldlength in the 2's complement format of weights of DCMA based on the proposed architecture is investigated. Moreover, 2's complement multipliers at input for DCMA are replaced by power-of-two multipliers. This is expected to reduce the resource for implementing the specific target hardware. The wordlenght of weights is varied to obtain the same target SINR obtained from the DCMA-based proposed architecture using 2's complement multipliers.

5.2 Delayed CMA for Adaptive Phased Array

5.2.1. Signal model and equations of DCMA

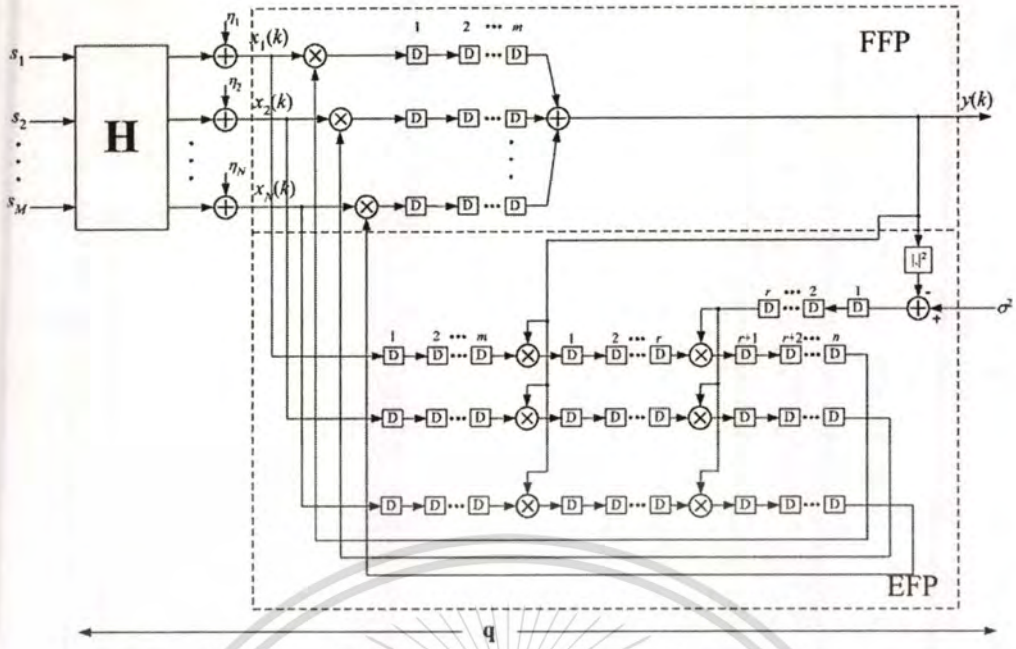


Fig. 5.1 Multiple-input multiple output model along with the DCMA configuration.

The multiple-input multiple output model along with the DCMA configuration is illustrated in Fig. 5.1. The received signal vector for CM receiver is given by

$$\mathbf{x} = \mathbf{H}\mathbf{s} + \eta, \tag{5.1}$$

where $\mathbf{x} = [x_0, x_1, \ldots, x_N]^T \in \Im^N$, $\mathbf{H} \in \Im^{N \times M}$ is the (unknown) array respond matrix, $\boldsymbol{s} = [s_1, s_2, \ldots, s_M] \in \Im^M$ is the vector of source signals, and $\eta = [\eta_1, \eta_2, \ldots, \eta_N] \in \Im^N$ is the additive noise. The received signals are fed to the DCMA. According to the theory of CMA addressed in chapter 2, Eq. (2.7) and (2.11) are keys for adaptation of the CMA array processing. Here, the algorithm is modified as DCMA. The DCMA can be described into two equations. Firstly, using vector notation, the delayed adaptive array output of DCMA can be written as

$$y(k\text{-}mD) = \mathbf{w}(k\text{-}mD)^H \mathbf{x}(k\text{-}mD) \tag{5.2}$$

where ^{H} is hermitian, $\mathbf{x}(k) = [\mathrm{x}_0(k)\ \mathrm{x}_1(k)\ \ldots\ \mathrm{x}_{N-1}(k)]^T$ is the complex signal received by i^{th} element of the antenna array in the k^{th} symbol interval $\mathbf{x}(k)$, $\mathbf{w}(k) = [\mathrm{w}_0(k)\ \mathrm{w}_1(k)\ \ldots\ \mathrm{w}_{N-1}(k)]^T$ is an adjustable complex weight, and m is the numbers of the delay unit D in

the array output equation normally referred to as the feed-forward path (FFP) of CMA. Secondly, the update weight vector is

$$\mathbf{w}(k+1)=\mathbf{w}(k)-4\mu\mathbf{x}(k-m\mathrm{D}-n\mathrm{D})y^{*}(k-m\mathrm{D}-n\mathrm{D})(|y(k-m\mathrm{D}-n\mathrm{D})|^{2}-\sigma^{2}) \quad (5.3)$$

where σ is the amplitude of the array output in the absence of interference, μ is a small step size constant and n is the numbers of the delay unit D in this equation normally referred to as the error-forward path (EFP) of CMA. Note that the delay D is inserted into both FFP and EFP since the DCMA is proposed in the context of an adaptive antenna. This is slightly different from the delayed adaptive filters that normally have the delay in only EFP [31].

According to Fig. 5.1, let us consider the total response vector [44] defined as

$$\mathbf{q} = \mathbf{w}^{H}\mathbf{H} \quad (5.4)$$

where **H** is an array response matrix. In this context, $\mathbf{q} = [Ae^{j\theta}\ 0]^{T}$ ($\mathbf{q} = [0\ Ae^{j\theta}\ ]^{T}$) corresponds to a situation that the antenna captures the desired (interference) user with scaling factor being A and overall phase shift being θ. The matrix **H** depends upon the antenna parameters and direction of the arrival of desired and interference signals, which can be easily determined.

5.2.2. Investigation of characteristic of DCMA via simulations with 2-element linear array antenna

A simulation was conducted to investigate the effect of inserting delay units into CMA array processing. In the simulation, the array antenna comprising two omni-directional antennas (N = 2) with half-wavelength spacing is considered. The desired and interference signals are π/4-QPSK-modulated. They arrive in the direction of 30° and 120°, respectively. Fig. 5.2 illustrates the SINR trajectories obtained from performing DCMA by varying the number of the delay unit. In the figure, the convergence rate of DCMA is fastest when the overall delay number being 0 (p = n+m = 0) and $\mu = 2^{-9}$. After increasing p to be 5, the convergence rate is relatively slow. Note that the DCMA has misconvergence when p is increased to be 10. This problem can be resolved by decreasing μ to be 2^{-10} as seen the SINR trajectory in the

figure. Although the DCMA with $p = 10$ and $\mu = 2^{-10}$ can converge to the desired signal, its convergence property is poor. According to the figure, it is noticed that introducing the delay into CMA degrades the convergence rate. Moreover, it is possible that the DCMA converges to the wrong solution (capturing interference) when p is much increased.

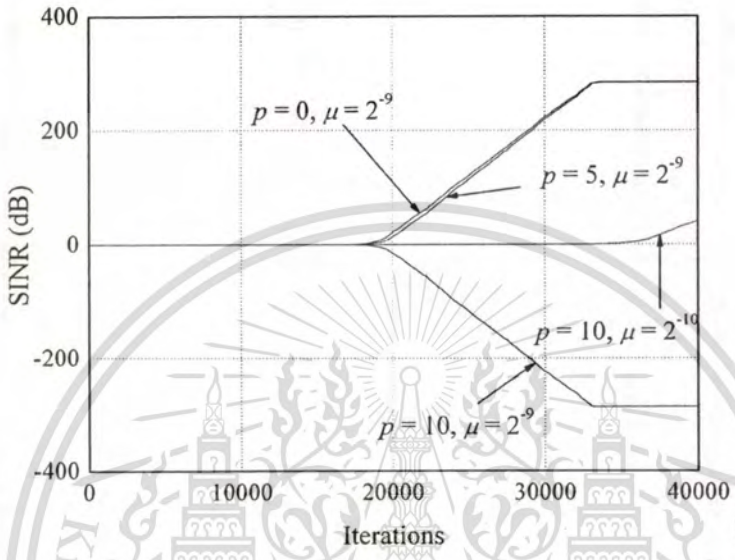


Fig. 5.2 SINR trajectories of the DCMA-based adaptive 2-element linear array antenna.

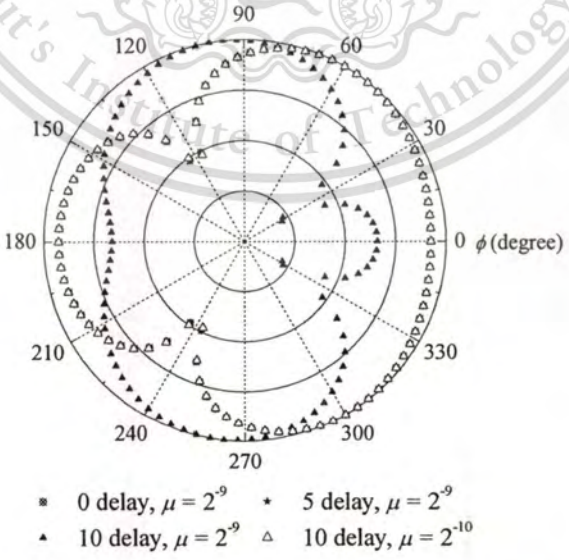


Fig. 5.3 Antenna pattern obtained from the DCMA-based adaptive 2-element linear array antenna at 40,000 iterations.

Moreover, careful investigation of the convergence property confirms that the desired (not interference) signal is captured. This is also validated by the final antenna pattern shown in Fig. 5.3. The figure shows the patterns obtained from the DCMA adaptive antenna by varying the number of delays in CMA processing. Obviously, at 40,000 iterations the main beam direction of the antenna pattern obtained from DCMA adaptive antenna with $p = 0$ and 5 and $\mu = 2^{-9}$ directs to 60°. The nullity appears at 120°. This clearly indicates that the interference signal was completely eliminated. As Fig. 5.2 shows that the antenna has misconvergence when $p = 10$ and $\mu = 2^{-9}$. Fig. 5.3 can also validate that the antenna eliminates the desired signal because its nullity appears at 30°. When decreasing μ to be 2^{-10}, the nullity of the antenna moves to 120°. This implies that inserting the delay into CMA can change the optimal solution of CMA. To avoid this problem, μ should be selected to be very small.

Besides, the signal constellation of the DCMA adaptive antenna is considered. Fig. 5.4 shows the signal constellation at the input of the DCMA processing. The received signal is not as concentrated as QPSK signal that it becomes a serious problem. Scattering of the received signal reveals that eye-pattern is closed. This is mainly due to the interference signal. With the presence of DCMA processing, signal constellation is improved as seen in Fig. 5.5-5.6. Fig. 5.5 (a) – (c) shows the output constellation of the DCMA adaptive antenna with $\mu = 2^{-9}$ after 30,000 iterations when $p = 0$, 5 and 10, respectively. The figures reveal that the eye-patterns are completely open. The eye-pattern in Fig. 5.5 (c) is opened, this constellation results from the interference signal as addressed above. The output constellation of the DCMA adaptive antenna with $p = 10$ and $\mu = 2^{-10}$ after 30,000 iterations is shown in Fig. 5.5 (d) in which it indicates that the eye-pattern is closed. This agrees with the SINR trajectories in Fig. 5.2 in which the SINR is about 0 dB between 0-35,000 iterations. With changing μ to be 2^{-10}, the output constellation of the DCMA adaptive antenna after 39,000 iterations is shown in Fig. 5.6. The figure obviously has an open eye-pattern.

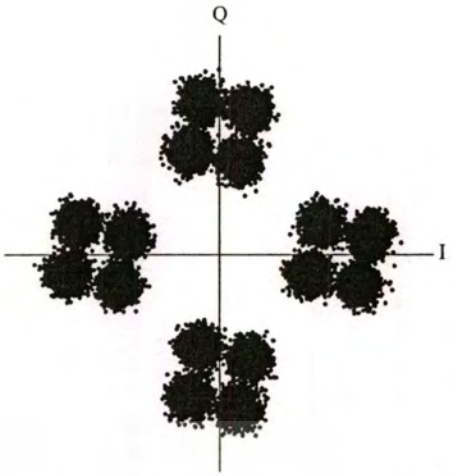


Fig. 5.4 Input constellation of DCMA.

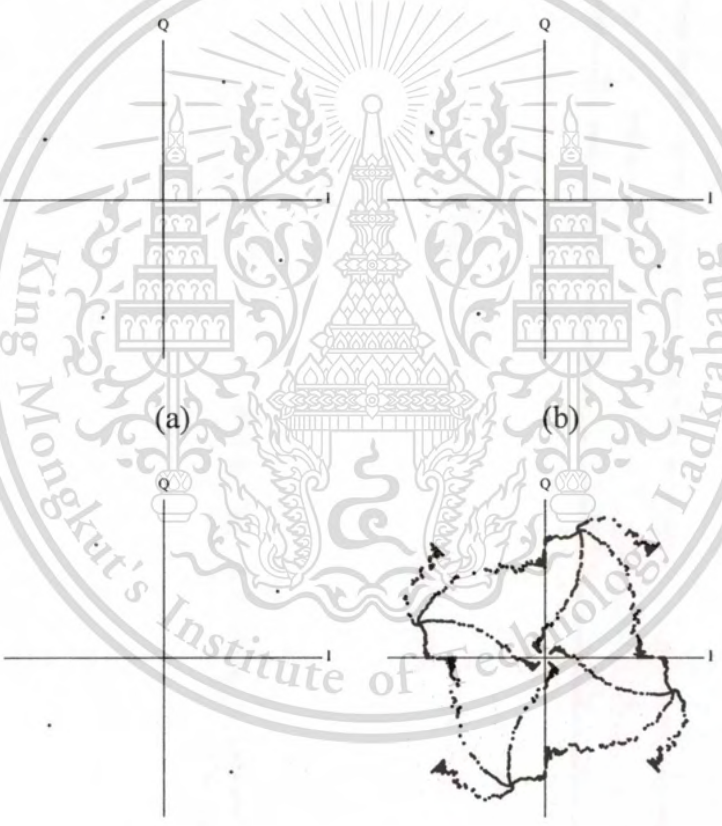


Fig. 5.5 Output constellation of DCMA after 30,000 iterations when (a) p = 0 $\mu = 2^{-9}$ (b) p = 5 $\mu = 2^{-9}$ (c) p = 10 $\mu = 2^{-9}$ (d) p = 10 $\mu = 2^{-10}$.

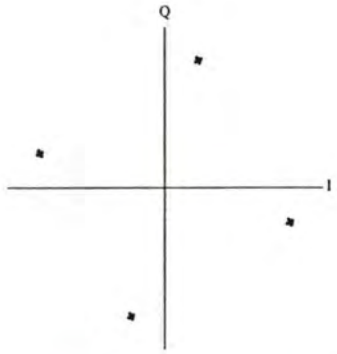


Fig. 5.6 Output constellation of DCMA after 39,000 iterations when p = 10 $\mu = 2^{-10}$.

To this end, the total respond vector **q** in Eq. (5.4) must be considered. The weight vector **w** obtained from performing DCMA and **H** depending on the antenna and the direction of arrival are substituted into Eq. (5.4) to find the vector **q**. Table 5.1 lists parameter values of DCMA at 40,000 iterations (after convergence). As seen in the table, when $p = 0$ and 5, **q** in polar form is approximately in the form of $[Ae^{j\theta}\ 0]^T$. The overall phase offsets are 17.7° and 14.6° when $p = 0$ and 5, respectively. These clearly imply that the desired signal is in fact captured with the overall phase offset being 17.7° and 14.6° agreeing with the output constellation in Fig. 5.5 (a) and (b), respectively. In contrast, the DCMA with $p = 10$ and μ 2^{-9} has misconvergence. Approximately, its total response vector is in the form of $[0\ Ae^{j\theta}]^T$ in which its phase offset is -14.4°. This also implies that the antenna captures the interference signal rather that the desired signal and involves the phase offset in the array output as seen in Fig. 5.5 (c). After decreasing μ to be 2^{-10}, the miscapturing can be resolved. The resulting solutions has **q** being $[1e^{j0.5243}\ 0e^{j1.4184}]^T$ nearly corresponding to $[Ae^{j\theta}\ 0]^T$ which indicate that the interference signal is eliminated.

Table 5.1 Parameter values of DCMA at 40,000 iterations.

p	μ	**w**	**q** (polar form)	phase shift
0	2^{-9}	$\begin{bmatrix} 0.5749+j0.1565i \\ -0.1565+j0.5749i \end{bmatrix}$	$\begin{bmatrix} 1e^{j0.3092} \\ 0e^{j0.7752} \end{bmatrix}$	17.7°
5	2^{-9}	$\begin{bmatrix} 0.5656+j0.1873i \\ -0.1873+j0.5656i \end{bmatrix}$	$\begin{bmatrix} 1e^{j0.2552} \\ 0e^{j0.7757} \end{bmatrix}$	14.6°

Table 5.1 (Cont.)

10	2^{-9}	$\begin{bmatrix} 0.5649\text{-}j0.1894 \\ 0.4382\text{-}j0.4036 \end{bmatrix}$	$\begin{bmatrix} 0e^{-j0.9039} \\ 1e^{-j0.2515} \end{bmatrix}$	-14.4°
10	2^{-10}	$\begin{bmatrix} 0.5907+j0.0255 \\ -0.0338+j0.5894 \end{bmatrix}$	$\begin{bmatrix} 1e^{j0.5243} \\ 0e^{j1.4184} \end{bmatrix}$	30.0°

5.2.3. DCMA-based proposed adaptive antenna

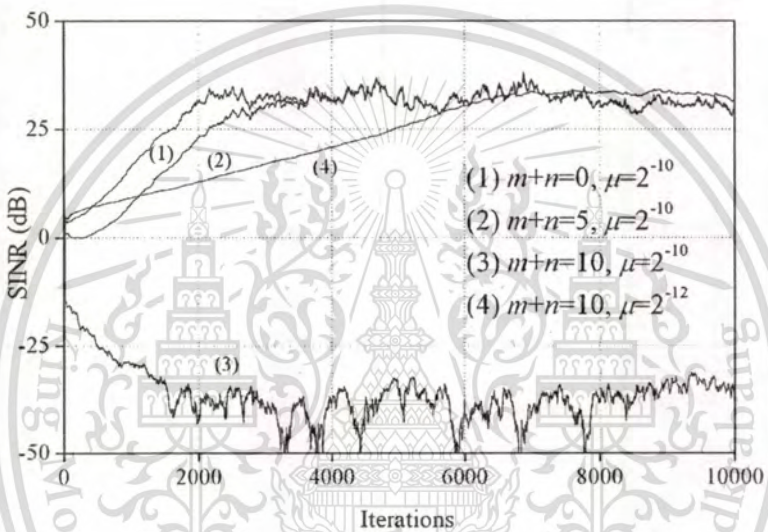


Fig. 5.7 SINR trajectories of the DCMA-based proposed adaptive antenna.

So far, we investigate the effect of the DCMA on the adaptive phased array antenna proposed in Chapter 3. The investigation is accomplished via simulations. In the simulations, the antenna is a flat four beam compact phased array antenna and its parameters were described in Chapter 2 and 3. The desired and interference signals were QPSK-modulated and propagated through an additive white Gaussian noise (AWGN) channel with 20 dB of signal-to-noise ratio (SNR). In practical wireless environment, the received power of the interference signal due to co-channel and multi-path fading is normally smaller than that of the desired signal. Therefore, in this simulation the power of the desired signal was set to be 3 dB stronger than that of the interference signal. The incident angles of the desired and interference signal were 30°and 120°, respectively. Following the approach in [38], [57], and [58] the main

beam of the antenna was switched to be 45° in order to improve convergence property. Fig. 5.7 illustrates SINR trajectories obtained from CMA with varying the numbers of the delay unit. The figure reveals that CMA with the overall delay numbers p being 5 converges slower than that without the delay. Clearly, the convergence property of CMA degrades when there are delays in CMA. Besides, the figure shows that SINR of CMA with p being 10 after convergence is negative. This implies that CMA captures the interference signal rather than the desired signal. When μ is decreased from 2^{-10} to 2^{-12}, CMA can reduce interference signal by capturing the desired signal as seen in the figure. According to Fig. 5.7, it is noted that the delay in CMA affects its convergence. To maintain the fact that CMA will capture the signal which has maximum power, μ should be selected to be very small.

The confirmation for the convergence property of DCMA is shown through the final antenna pattern shown in Fig. 5.8. In the figure, the main beam and nullity of the antenna based on DCMA with $\mu = 2^{-10}$ and p = 0 (non-pipelining) are respectively in the direction of 32° and 118°. When $\mu = 2^{-10}$ and $p = 5$, the main beam directs to 29° and the nullity appears at 118°. These antenna patterns are almost identical and reveal that the antenna eliminates the interference signal. The SINRs obtained from the DCMA-based adaptive antenna with $\mu = 2^{-10}$ and $p = 0$ and 5 are 30.94 dB and 31.16 dB, respectively. The antenna with $\mu = 2^{-10}$ and $p = 10$ captures the interference signal as the figure shows that its nullity is in the direction of 31°. Its SINR is -35.53 dB corresponding to the SINR trajectory shown in Fig. 5.7 in which SINR has negative values. When μ is decreased to be 2^{-12}, the antenna changes to capture the desired signal as seen in the figure where its main beam directs to 33° and nullity appears at 119°. The SINR is 31.18 dB.

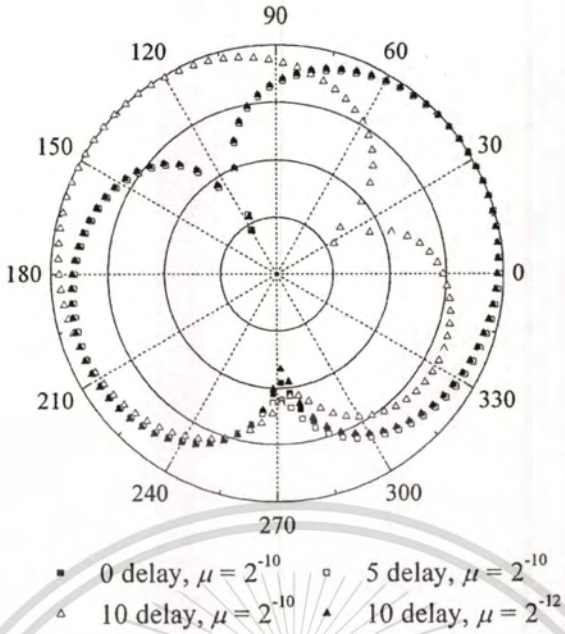


Fig. 5.8 Antenna pattern of the DCMA-based proposed adaptive antenna at 10,000 iterations.

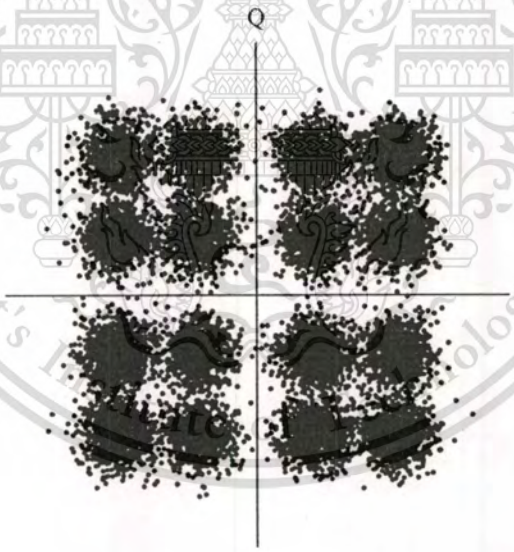


Fig. 5.9 Input constellation of the DCMA-based proposed adaptive antenna.

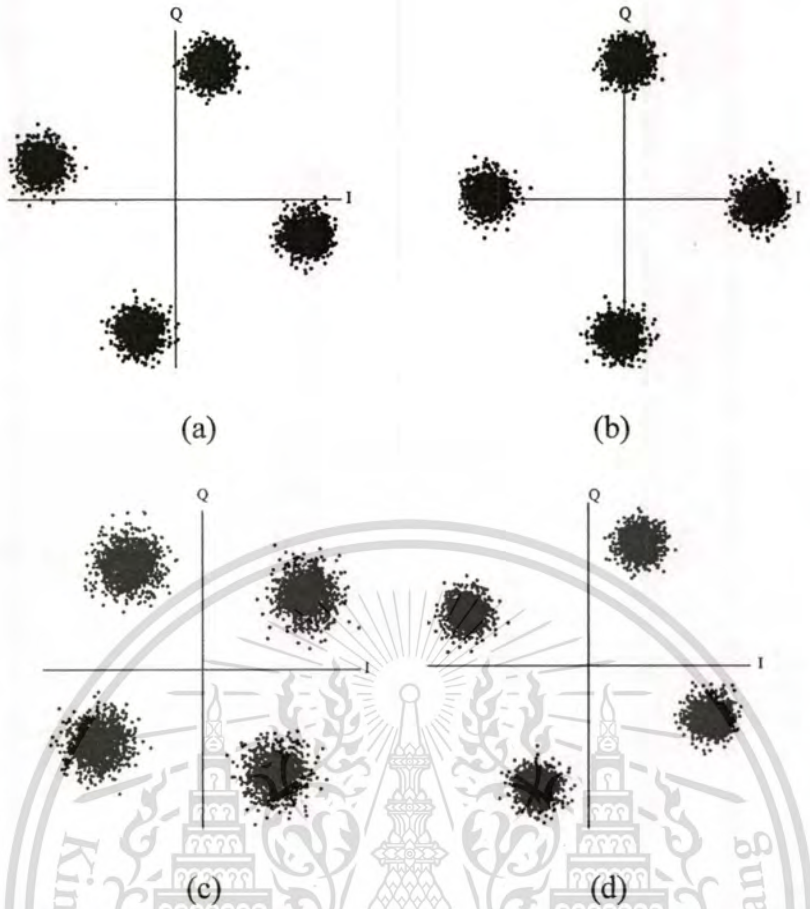


Fig. 5.10 Output constellation of the DCMA-based proposed adaptive antenna after 5,000 iteration when (a) p = 0 $\mu = 2^{-10}$ (b) $p = 5$ $\mu = 2^{-10}$ (c) $p = 10$ $\mu = 2^{-10}$ (d) $p = 10$ $\mu = 2^{-12}$.

The signal constellations of the DCMA-based proposed adaptive antenna are illustrated in Fig. 5.9 and 5.10. Fig. 5.9 shows the input constellation for DCMA array processing. The signal is not as concentrated as QPSK signal. Scattering of the signals reveals that the eye-pattern is closed. This is mainly due to the interference signal and noise. The signal constellation is improved when DCMA is performed as shown in Fig. 5.10. All of the output constellations obtained from performing DCMA after 5,000 iterations are opened. However, scattering of the output constellation shown in Fig. 5.10 (c) is due to the interference signal as addressed above.

Similar to consideration of the total respond vector in Section 5.2.2, Table 5.2 lists the parameter values obtained from performing DCMA and calculating Eq. (5.4). The weight vectors **w** after convergence (at 10,000 iterations) of CMA with μ being 2^{-10} and p being 0 and 5 are $[0.0438+j0.0205 \quad 0.2328-j0.3118 \quad 0.1440-j0.1775 \quad 0.4299+j0.0327]^T$ and $[0.0044+j0.0046 \quad 0.1596-j0.3363 \quad 0.1411-j0.2205 \quad 0.4277-$

$j0.0453]^T$, respectively. Although CMA with and without the delay capture the desired signal (not interference), their weight vectors after convergence are different. Thus, note that CMA with the different delay numbers may have different solutions.

By substituting **w** and **H** in Eq. (5.4), the total response vectors **q** are $[0.8496+j0.4876 \ -0.0212-j0.0179]^T$ and $[0.7179+j0.6673 \ -0.0160-j0.0219]^T$ for p being 0 and 5 in CMA, respectively. Equivalently, these correspond to **q** being $[0.9796e^{j0.5210} \ 0.0278e^{-j2.4401}]^T$ and $[0.9802e^{j0.7489} \ 0.0271e^{-j2.2007}]^T$ in polar form and clearly implies that the desired signal is in fact captured with the overall phase offset being 29.8° and 42.9°. The overall phase offset obtained from both cases will produce the phase offset in the array output (I and Q) of CMA as seen in Fig. 5.10. The phase ambiguity of the output can be typically resolved through the use of a unique word periodically inserted in the data stream, for example, using Turbo code or Reed-Solomon block code [59].

Finally, we determine the solution for $p = 10$ with $\mu = 2^{-10}$ and 2^{-12}. The weight vectors **w** after convergence (at 10,000 iterations) of CMA with $p = 10$ and μ being 2^{-10} and 2^{-12} are $[-0.2101+j0.3936 \ -0.3407-j0.2039 \ 0.0363+j0.0647 \ 0.5568-j0.0407]^T$ and $[0.0658+j0.0335 \ 0.2751-j0.2891 \ 0.1405-j0.1521 \ 0.4216+j0.0800]^T$, respectively. From these weight vectors, **q** is $[0.0229e^{j1.8130} \ 1.3685e^{-j1.7463}]^T$ and $[0.9794e^{j0.3828} \ 0.0270e^{-j2.7098}]^T$ and implies that the CMA with $p = 10$ captures the interference signal when μ being 2^{-10} while the desired signal when μ being 2^{-12}. Note that this capture is in good agreement with the SINR trajectories in Fig. 5.7-5.8.

Table 5.2 Parameter values of the DCMA-based proposed adaptive antenna at 10,000 iterations.

p	μ	**w**	**q** (polar form)	phase shift
0	2^{-10}	$\begin{bmatrix} 0.0438+j0.0205 \\ 0.2328-j0.3118 \\ 0.1440-j0.1775 \\ 0.4299+j0.0327 \end{bmatrix}$	$\begin{bmatrix} 0.9796e^{j0.5210} \\ 0.0278e^{-j2.4401} \end{bmatrix}$	29.8°

Table 5.2 (Cont.)

5	2^{-10}	$\begin{bmatrix} 0.0044+j0.0046 \\ 0.1596-j0.3363 \\ 0.1411-j0.2205 \\ 0.4277-j0.0453 \end{bmatrix}$	$\begin{bmatrix} 0.9802e^{j0.7489} \\ 0.0271e^{-j2.2007} \end{bmatrix}$	42.9°
10	2^{-10}	$\begin{bmatrix} -0.2101+j0.3936 \\ -0.3407-j0.2039 \\ 0.0363+j0.0647 \\ 0.5568-j0.0407 \end{bmatrix}$	$\begin{bmatrix} 0.0229e^{j1.8130} \\ 1.3685e^{-j1.7463} \end{bmatrix}$	-100.1°
10	2^{-12}	$\begin{bmatrix} 0.0658+j0.0335 \\ 0.2751-j0.2891 \\ 0.1405-j0.1521 \\ 0.4216+j0.0800 \end{bmatrix}$	$\begin{bmatrix} 0.9794e^{j0.3828} \\ 0.0270e^{-j2.7098} \end{bmatrix}$	21.9°

5.3 Proposed Efficient Architecture for DCMA Array

Delay in the weight adaptation of CMA has only a slight influence if the step size is very small within the certain bound in [18]. In this section, we propose an efficient architecture by introducing pipelining. The proposed architecture can calculate in parallel. By applying DCMA, pipelining registers can be inserted into the FFP and EFP in order to reduce the critical path in the loop of the recursive adaptive system. Without pipelining, the critical path of a processing unit can be calculated by summation of processing times of all operators. Fig. 5.11 indicates the possible pipelining paths in the CMA adaptive array processing unit. The pipelined DCMA adaptive array is implemented by inserting pipelining registers along the FFP and EFP. Here, a register inserted for implementing the processing unit is a delay in processing of the recursive algorithm. As seen in the figure, it is possible that the delay D is inserted after each operator namely an adder and multiplier. Thus, the critical path of the pipelined DCMA adaptive array processing unit is equal to the processing time of an adder or multiplier. Although the pipelining technique can reduce the critical path of the system, it increases hardware complexity resulting from an increase of registers. In addition, it increases output latency which degrades the

convergence property of CMA being shown in the investigation of wordlength effect. After the proposed architecture, we structure the pipelined CMA by using a fixed-point and power-of-two arithmetic. The weights of an adaptive algorithm being implemented on FPGA or special-purpose hardware will be represented by a finite number of bits. Thus, the pipelined CMA with a small number of bits of weights is required. We investigate the wordlength effect on the CMA adaptive phased array using the proposed architecture along with fixed-point and power-of-two operators.

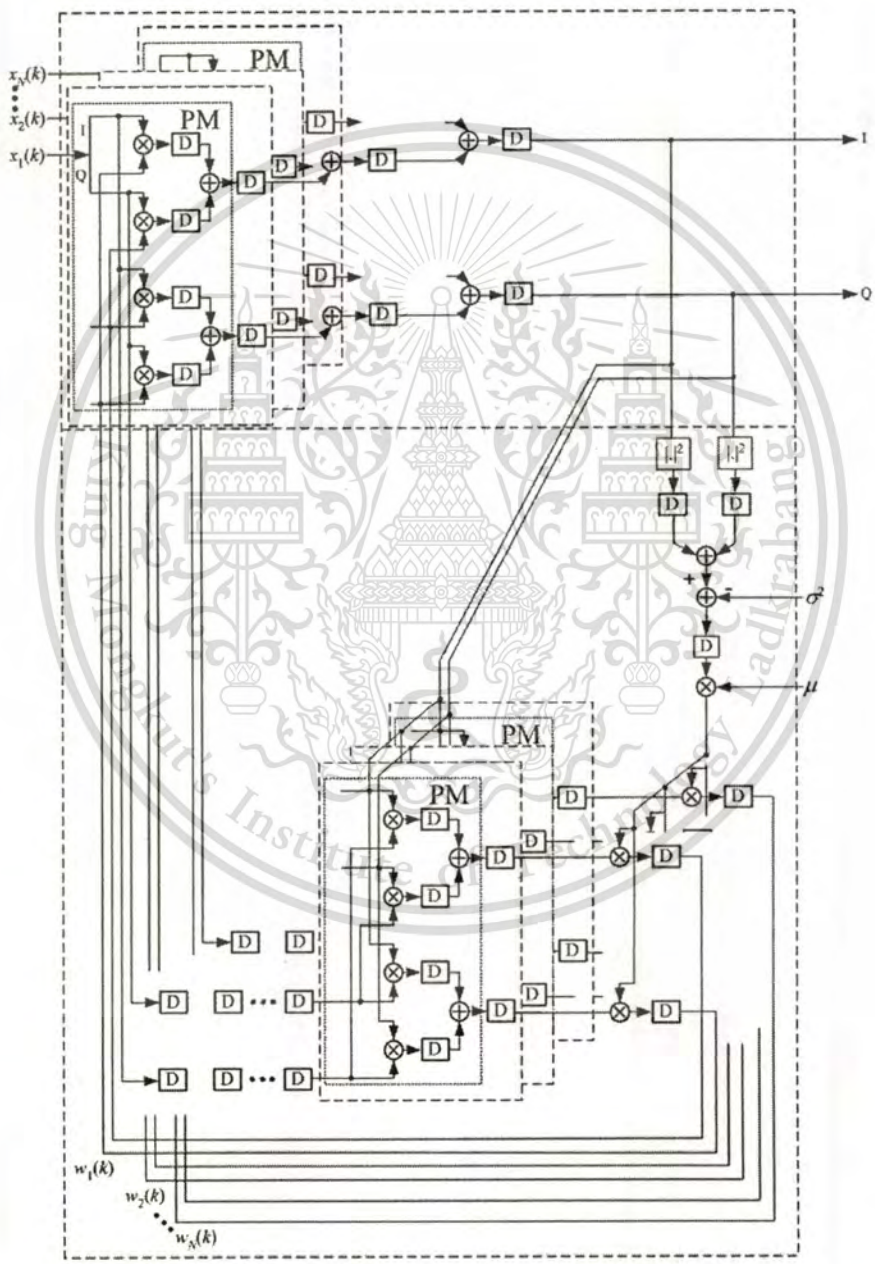


Fig. 5.11 Architecture of CMA adaptive processing unit with the possible pipelining paths.

5.3.1. Feed forward part

According to Eq. (5.2), the FFP mainly comprises the multiplication and accumulation. The inner product can be calculated by multiplication between a conjugated complex weight and a complex signal of an individual antenna element. The outputs from each complex multiplication are accumulated together. The complex multiplication can be represented by a processing module (PM). The PM structure can be constructed via the multiplication between two complex number, for example, $(a+jb)^*(c+jd) = (ac+bd) + j(ad-bc)$. Fig. 5.12 illustrates the PM structure comprising four multipliers and two adders. The output of the PM is distinguished into two parts: real and imaginary parts. Without pipelining, the critical path is processing time from input to output of the PM namely $T_m + T_a$ where T_m and T_a is a processing time for a multiplier and an adder, respectively. In the PM with pipelining, a register (delay unit D) is inserted after each multiplier as seen in Fig. 5.12, to reduce critical path from $T_m + T_a$ to T_m. Here, we assume that T_m is about $3T_a$ [31].

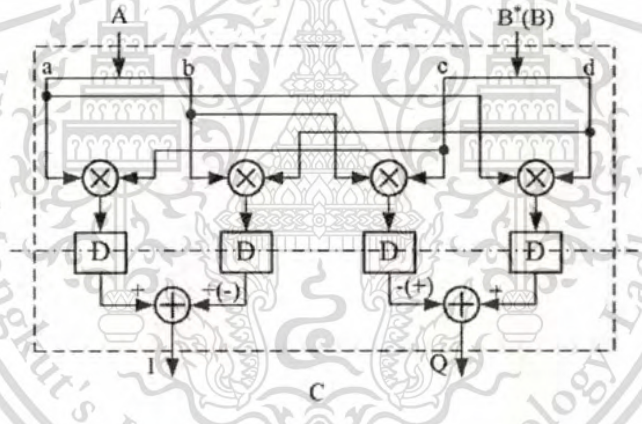


Fig. 5.12 Processing module (PM) structure.

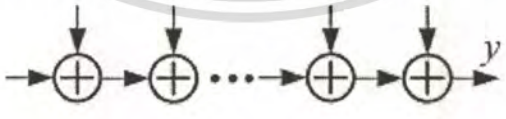


Fig. 5.13 Structure of the direct-form addition.

The FFP can be constructed by using the PMs. The direct-form addition of the FFP is impractical for the high-speed architecture as the accumulation of the products requires at least $4N$-2 adders for both real and imaginary parts, and cannot be computed in parallel whereas the multiplication can, where N is the number of an antenna element. The structure of the direct-form addition is shown in Fig. 5.13.

Pipelining is possible to enable concurrent operation but the number of delay units for pipelining increases proportionally with that of the antenna element.

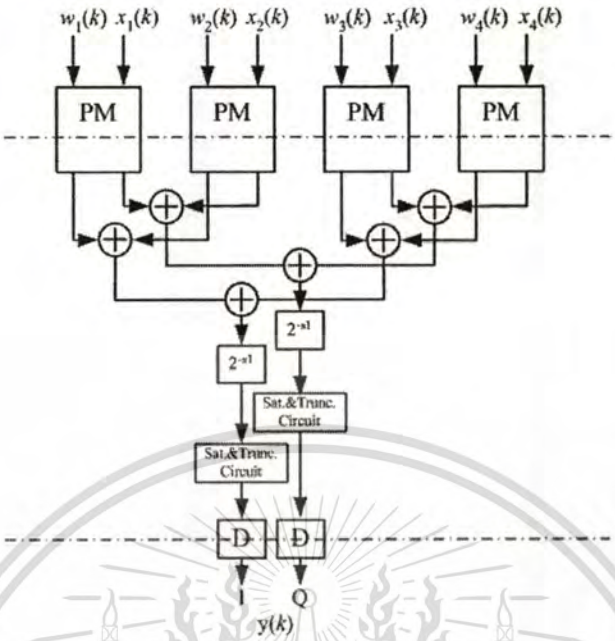


Fig. 5.14 Structure of feed-forward path of the proposed architecture of DCMA.

The degradation of the convergence property, which is due to an increase in the number of the pipelining delays in both of the FFP and EFP, is still the main concern of the pipelined DCMA architecture. In this thesis, the conventional structure of direct-form addition is replaced by the binary-tree adder structure as seen in Fig. 5.14. This can reduce the number of latency (mD) from $2N$-1 to $\log_2 N$+1. Here, N is four. Thus, the mD which includes the delay after complex multiplications and additions in the FFP should be equal to 4D. However, since T_m is assumed to be about $3T_a$, inserting the delay after every operator is not necessary. Therefore, the delay unit D is inserted only after each multiplier and adder at the final state of tree as seen in Fig. 5.12 and 5.14. In the FFP (real and imaginary parts), the latency and the number of registers can be respectively reduced from 7D to 2D by using binary-tree adder structure and from thirty to eighteen by omitting delays after some adders. Note that the critical path in the FFP is also T_m.

Two techniques namely fixed-point and power-of-two operations are employed to reduce the use of resource for hardware implementation. With the binary-tree adder structure with and without power-of-two multipliers, the numbers of bits required for the array output of CMA respectively are

$$b_{IQ} = b_I + b_w + \log_2 N + 1 \tag{5.5}$$

and

$$b_{IQ} = b_I + b_w + \log_2 N - 1 \tag{5.6}$$

where b_I and b_w respectively are wordlengths of inputs and weights. Extending the precision of a word can always be accomplished with more bits, but practical limitations with this approach are encountered. A saturation and truncation circuit dealing with the precision of a fixed-point arithmetic is used to limit the wordlength. It is vital to ensure that the signal is in the proper range. Therefore, the scaling inserted by the power-of-two multiplier before the saturation and truncation circuits is necessary to prevent overflow and underflow phenomena before rounding. The scaling with 2^{-s1} ensures that the array output is in the dynamic range of a desirable wordlength. However, this is not necessary since the fixed-point number can represent expected signals sufficiently.

5.3.2. Error forward part

Similar to construction of the FFP, the EFP (Eq. 5.3) is constructed by using a PM structure as seen in Fig. 5.15. In the part of $\mathbf{x}(k)y^*(k)$, the PM with pipelining (Fig. 5.12) is used to construct. To obtain the concurrent operation, the input signal $\mathbf{x}(k)$ is delayed with registers of 2D as the array output obtained from the FFP has latency being 2. The critical path in the part of $\mathbf{x}(k)y^*(k)$ is T_m. Simultaneously, the part of $|y(k)|^2-\sigma^2$ is provided. The construction of $|y(k)|^2$ should be conducted by using the look-up table of squarers rather than multipliers of PM. It can reduce the use of resource of hardware implementation [60]. However, the critical path of overall architectures must be limited to be T_m. Thus, the delay unit D is inserted after each squarer. This does not only maintain the critical part which is less than T_m but also provide the concurrent outputs of $\mathbf{x}(k)y^*(k)$ and $|y(k)|^2-\sigma^2$. The operation of $\mathbf{x}(k)y^*(k)$ and $|y(k)|^2-\sigma^2$ is multiplication between complex signals and real signals. Thus, only multipliers (not include adders) is required. Similarly, the delay unit D is inserted after the multiplication of $\mathbf{x}(k)y^*(k)$ and $|y(k)|^2-\sigma^2$. Here, the critical path is T_m+T_a because of addition and subtraction in part of $\mathbf{x}(k)y^*(k)$ and $|y(k)|^2-\sigma^2$. The critical part is limited to be T_m by inserting the delays D into the interconnection between adders in part of $\mathbf{x}(k)y^*(k)$ and $|y(k)|^2-\sigma^2$ and multipliers of $\mathbf{x}(k)y^*(k)$ and $|y(k)|^2-\sigma^2$ as seen in

Fig. 5.15 Referring to Fig. 5.15, the number of delays in the EFP (*n*D) is 3.

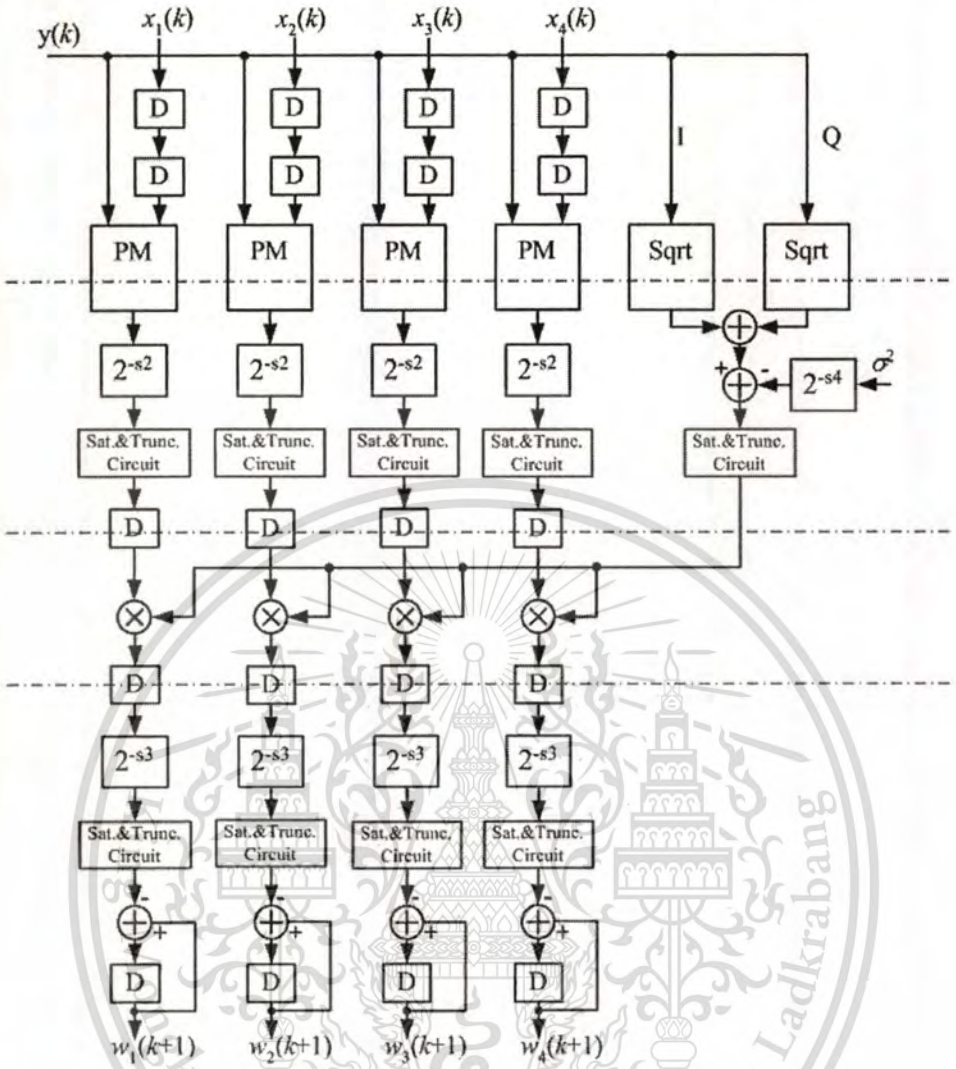


Fig. 5.15 Structure of error feed forward part of the proposed architecture of DCMA.

With extending the wordlength due to multiplications and accumulations, the saturation and truncation circuits were inserted before each multiplier. The number of bits required for inputs of multiplier in an internal architecture was chosen to be equal to that of weights. It ensures that the gradient estimation is in the range of the representation of a number. Before the saturation and truncation circuits, the scaling was used to prevent the overflow phenomena. However, if the scaling incorporating the step size constant μ with very small value is used at only one part of CMA, the underflow is possible to be encountered easily. Thus, dividing the scaling into 2^{-s2} and 2^{-s3} inserted after $\mathbf{x}(k)y^*(k)$ and $\mathbf{x}(k)y^*(k)(|y(k)|^2-\sigma^2)$ can mitigate the problem but does not nearly impact on the performance of CMA. Moreover, the step size constant μ of power-of-two is used to reduce the scaling multiplication in CMA to a simple shift

operation. Obviously, it does not only incorporate the step size constant μ but also can prevent underflow and overflow resulting from the multiplications and accumulations. Furthermore, the scaling with 2^{-4} is necessary if the power of the dummy reference signal (σ^2 is constant) is not in the same range of that of the desired signal. Due to the scaling, the processing time of multiplication between the step size constant μ and another signal can be ignored. In the ultimate architecture of CMA, the critical path of overall architecture of CMA is T_m and the number of latency in the FFP and EFP of CMA is 2 and 3, respectively.

5.3.3. Effect of finite wordlength of weights

Let us consider the wordlength effect of weights on the pipelined CMA being implemented by using a fixed-point arithmetic. Eight-bit inputs and outputs (I/Os) were selected for wireless communication applications [48]. It is well-known that pipelining used in a recursive algorithm degrades its convergence property. Thus, the proposed technique is not only the efficient architecture achieving a high throughput, but also the pipelined CMA using beam-switching initialization which can resolve its slow convergence property [38], [57], and [58].

Simulations of the adaptive processing unit of the proposed DCMA-based pipelined architecture with different wordlengths were conducted. The SINR obtained from the pipelined CMA-based adaptive antenna using a fixed-point arithmetic with and without power-of-two multipliers was examined. The desired and interference signals were QPSK-modulated and propagated through an AWGN channel with 20 dB of SNR. In practical wireless environment, the received power of the interference signal due to co-channel and multi-path fading is normally smaller than that of the desired signal. Therefore, the power of the desired signal was set to be 3 dB stronger than that of the interference signal. The incident angles of the desired and interference signal were 30°and 120°, respectively. The antenna was a flat four-beam compact phased array antenna and its parameter was described in Chapter 2 and 3.

Fig. 5.16 illustrates the SINR trajectories of the pipelined CMA-based adaptive antenna using a fixed-point arithmetic with the proposed architecture by varying wordlength of weight vectors. In this work, the target SINR is more than 20 dB. For 6-bit weight vectors, the average SINR after convergence of CMA is smaller than 20 dB. This means that the wordlength is not enough. The wordlength should be

therefore increased. The figure reveals that the 5D pipelined CMA has a better convergence property. According the figure, after 2,000 iterations, the 5D pipelined CMA with 10-bit weights has a relative high SINR (above 20 dB) compared with that with other wordlengths. This indicates that the wordlength of weight vectors is enough for an adaptation.

In order to reduce the use of resource for efficient-hardware implementation, fixed-point multipliers at inputs for operation of $x_i(k)y^*(k)$ and $w_i(k)x_i(k)$ are replaced by power-of-two multipliers. The total number of replaced multipliers is thirty-six. One of the inputs of the multiplier must be in a power-of-two format. In this paper, input signals of CMA are arranged to be a power-of-two format with sign-magnitude representation. On the other hand, weights are arranged to be a fixed-point format with 2's complement representation since it requires high resolution. The reason of choosing the 2's complement representation is that its adder usually uses a small resource. The wordlenght effect on the pipelined CMA-based adaptive antenna using a fixed-point arithmetic with power-of-two multipliers was investigated. The SINR trajectories of the proposed adaptive antenna with different wordlengths are illustrated in Fig. 5.17. Clearly, the SINR increases proportionally with wordlength of weight vectors. However, this will be saturated with one wordlength. To obtain SINR above 20 dB after convergence, the wordlengths of the pipelined CMA-based adaptive antenna with and without power-of-two multipliers is selected to be 10 and 12, respectively.

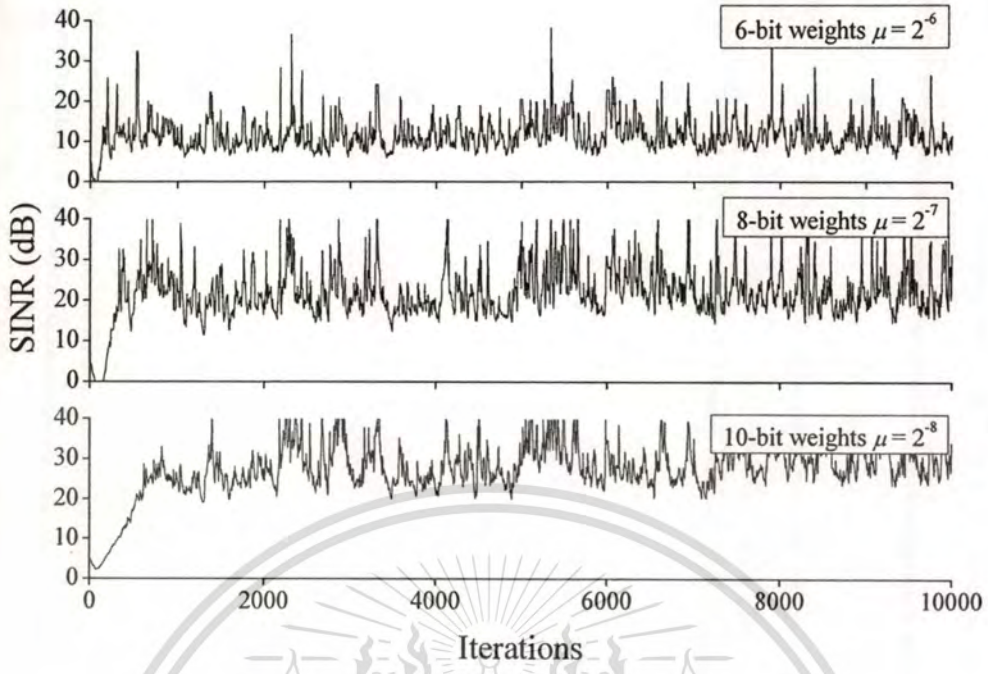


Fig. 5.16 SINR trajectories of the 5D pipelined CMA using a fixed-point arithmetic.

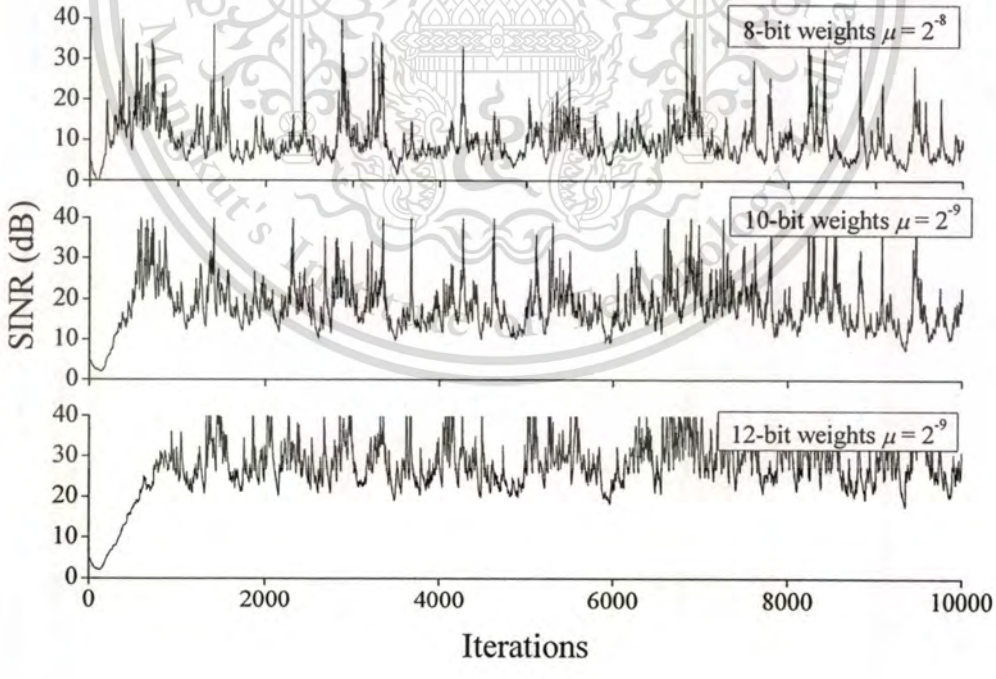


Fig. 5.17 SINR trajectories of the 5D pipelined CMA using the power-of-two arithmetic at inputs of CMA.

5.4 Summary

The DCMA-based efficient architecture of the adaptive array processing unit has been presented in this chapter. The signal model and equations of DCMA were described in the context of array signal processing. The characteristic of DCMA was investigated via simulations of the proposed phased array antenna and 2-element linear array antenna. The simulation results show that the present of delays in CMA degrades the convergence rate of CMA. In addition, inserting delay units into CMA can change the final solution obtained from performing DCMA. The efficient architecture for the DCMA array was proposed to achieve its critical path being T_m. Two techniques of fixed-point and power-of-two operations were introduced to reduce either the power consumption or the use of resource for hardware implementation. Finally, the effect of finite wordlength of weights is investigated to achieve the relative high SINR. The 10 and 12-bit weights of fixed-point and power-of-two operations were chosen, respectively.

CHAPTER 6
IMPLEMENTATION OF CMA ARRAY PROCESSING UNIT WITH EFFICIENT ARCHITECTURE

6.1 Introduction

In this chapter, the DCMA array processing unit with the proposed pipelined architecture will be implemented on FPGA by using fixed-point and power-of-two operations. The basic component circuits including adders, multipliers, scaling, and registers will be described. The use of resource for hardware implementation of the fixed-point and power-of-two multipliers will be considered. The test of FPGA of the DCMA adaptive array processing unit presented with the proposed architecture in Chapter 3 will be conducted. The testing results will be shown and compared with the simulation ones.

6.2 Fixed-point and Power-of-two Representation

To reduce the power consumption, two techniques: fixed-point and power-of-two representation, were exploited for the proposed pipelined architecture. The numbers which are in fixed-point and power-of-two formats can be represented as

$$A=a_{W-1}.a_{W-2}\cdots a_1a_0 \tag{6.1}$$

where the bits a_i, $0 \le i \le W-1$, are either 0 or 1, and the most significant bit (msb) a_{W-1} is the sign bit with value of 0 denoting positive number and 1 denoting negative number. With 2's complement representation, the value of the fixed-point number is in the range of $[-1, 1-2^{-W+1}]$ and is given by

$$A=-a_{W-1}+\sum_{i=1}^{W-1} a_{W-1-i}2^{-i}\,. \tag{6.2}$$

The main advantage of 2's complement arithmetic is the ability to generate a correct final result in spite of intermediate overflow [61]. Thus, in a chain of additions and subtractions, the final result is correct if it is known to lie in the range $[-(1-2^{-W+1}), 1-2^{-W+1}]$ even though intermediate operations lead to overflow.

With sign-magnitude representation, the value of the power-of-two number is in the set of $[-2^{a_{W-2}}, -2^{a_{W-3}}, \ldots, -2^{a_0}, 2^{a_0}, 2^{a_1}, \ldots, 2^{a_{W-3}}, 2^{a_{W-2}}]$ and is given by

$$A=(-1)^{a_{W-1}} \times 2^{a_{We}} \tag{6.3}$$

where a_{We} is the order of significance of bit 1 not including a sign bit.
The main advantage of the power-of-two arithmetic is reducing the used resource for implementing a multiplier. Besides, it is easy to determine bit 1 of input signals in order to shift bits of weights also.

6.3 Component Circuits

The components used to construct the 5D pipelined architecture of the DCMA adaptive array processing unit was implemented on the Xilinx Virtex-E XCV400E FPGA. The main components are briefly described as follows.

6.3.1. Multipliers

Fixed-point and power-of-two multipliers were used to construct the 5D pipelined CMA. Generally, to construct a power-of-two multiplier, one of the input signals is in the power-of-two format and another is in the common format such as 2's complement and sign-magnitude formats. In this thesis, a power-of-two multiplier was used to replace a fixed-point multiplier for operation between input signals $\mathbf{x}(k)$ and others as described in Chapter 4. By replacing the multiplier, all signals except input signals in CMA processing were also represented by fixed-point numbers with 2's complement representation. The input signals were arranged in power-of-two formats with sign-magnitude representation. Power-of-two multiplication is shifting bit of one of the inputs of the multiplier by determining bit 1 in another (power-of-two format) of the inputs of the multiplier. The number of shift depends on the position of bit 1 in the power-of-two number. In the power-of-two arithmetic, if one of the inputs of the multiplier is negative, the sign of the output will oppose that of another input. After bit-shifting, the output must be converted by inverting each bit and then adding 1 at the least significant bit (LSB). Adding 1 leads to a large used resource. Here, it is not taken into implement in the power-of-two multiplier since the LSB of product will be truncated to reduce the wordlength. Thus, there is a small error due to not adding 1 when the sign bit of the input signal (power-of-two number) is negative.

Fig. 6.1 shows the use of resource of a fixed-point and power-of-two multiplier along with registers. The multipliers were implemented on the Xilinx Virtex-E XCV400E FPGA device. According to Chapter 4, since 8-bit I/Os of the DCMA array processing unit were selected, one of the inputs of multipliers has wordlength being 8. The wordlength of another inputs being fed for weight vectors was varied. At all of the wordlengths of inputs of the multipliers, the figure indicates that the total equivalent gate count of a fixed-point multiplier is larger than that of a power-of-two multiplier. The used resource increase propotionaly with the wordlength of the input of a multiplier. In addition, when the wordlength of the input of a multiplier is increased, an increase in the used resources of the power-of-two multiplier is higher than that of the fixed-point multipliers. Referring to Fig. 5.16-5.17, when the wordlenth of weight vectors with the architectures using fixed-point and power-of-two operations is respectively selected to be ten and twelve, the total equivalent gate count for a 8x10-bit fixed-point and 8x12-bit power-of-two multiplier was 1,225 and 654, respectively. The consumed resources for a multiplier can be reduced with the percentage of 46.6. This implies that the consumed resource of FPGA implementation of the 5D pipelined CMA adaptive processing unit can be significantly reduced by replacing the fixed-point multiplier by the power-of-two multiplier. It is more efficient if the more accuracy (longer wordlength of weights) of CMA is required.

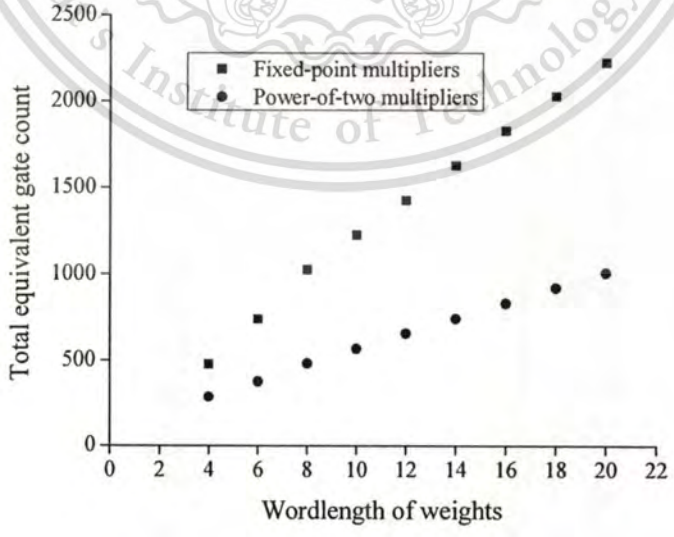


Fig. 6.1 Total equivalent gate count of fixed-point and power-of-two multipliers with various wordlength of its input.

6.3.2. Adders

Adders in the proposed architecture using fixed-point and power-of-two multipliers were parallel operations with carry ripple technique with 2's complement representation which is simple and uses small resource for hardware implementation.

6.3.3. Registers

A register is used to store signals and operate as delay in CMA.

6.3.4. Saturation and Truncation Circuits

A saturation and truncation circuit dealing with the precision of the arithmetic is used to limit the wordlength. Fig. 6.2 illustrates a saturation and truncation circuit. Since the input signals for CMA is limited by combination of the desired and interference signals, weights without a sign in CMA should be 1 or less for stable operation. Bits which are on left hand side of radix point (not including sign bit) represent 1 or more. These bits will be fed into the AND and OR gates to determine overflow and underflow phenomena. With a sign bit being 1 (negative number) when the output of the AND gate is zero, the input will underflow. The output of the saturation and truncation circuit is therefore selected to be a minimum value with 2's complement representation for its expected bit numbers. On the contrary, with a sign bit being 0 (positive number) when the output of the OR gate is one, the input will then overflow. Thus, the output is the maximum value. When the input does not overflow and underflow, it is truncated and then selected for output.

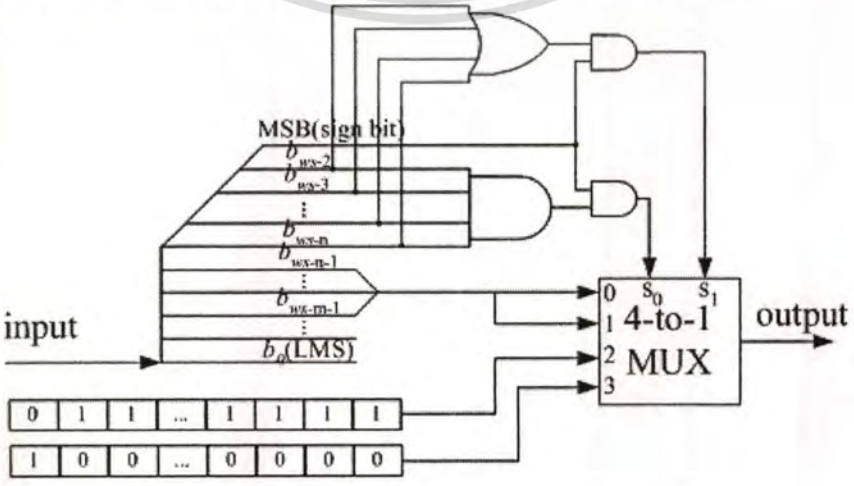


Fig. 6.2 Saturation and truncation circuit.

6.3.5. Scaling Circuits

A scaling circuit providing the step size constant μ is shifting bit to the right hand side. This scaling is used in the algorithm to reduce the resource for scaling multiplication to a simple shift operation.

6.4 DCMA-Based Adaptive Array Processing Unit on FPGA

The implementation of the adaptive array processing unit with the DCMA-based pipelined architecture was realized. According to investigation of wordlength effect in Chapter 4, it is shown that the fixed-point and power-of-two formats can represent the adjustable weights precisely. 10-bit and 12-bit weights are respectively chosen for the fixed-point and power-of-two arithmetic. The increased wordlenghts can result in the more accurate adaptation and the high SINR at the expense of size (resource) and longer critical paths. The basic components addressed in Section 6.3 were utilized to construct the proposed DCMA-based pipelined architecture for the adaptive processing unit. The circuit of the unit along with these components was synthesized by using VHDL and then implemented on an FPGA device. To verify that the implemented unit can operate correctly, it must be tested.

6.4.1. Testing System

Fig. 6.3 illustrates the testing system of the proposed DCMA-based pipelined architecture of the adaptive processing unit. The input signals are required to test the unit. With the limitation of resource of FPGA, only 2048 data samples of the input signals are generated by the simulation with a high-level language and then downloaded into RAMs on FPGA. The control circuit was created to enable CMA and provide the address for reading RAMs. To start control, the pull-down circuit was employed. Here, the master clock of operation is 16 MHz. The output signals obtained from performing the unit were collected by using a program of Chip Scope Pro. RAMs were automatically generated to interface between the implemented unit and PC. Interfacing between the unit and PC is achieved via a parallel port.

The proposed adaptive array antenna (in Chapter 3) was utilized to generate the input signals $x(k)$. The desired and interference signals were $\pi/4$-QPSK and

propagated through an AWGN channel. The power of the desired signal was set to be 3 dB stronger than that of the interference. Following Chapter 3, the main beam of the antenna was initially in the direction of $\phi_0 = 45°$. The input signals for CMA processing was obtained from these conditions, and then fed to the implemented unit.

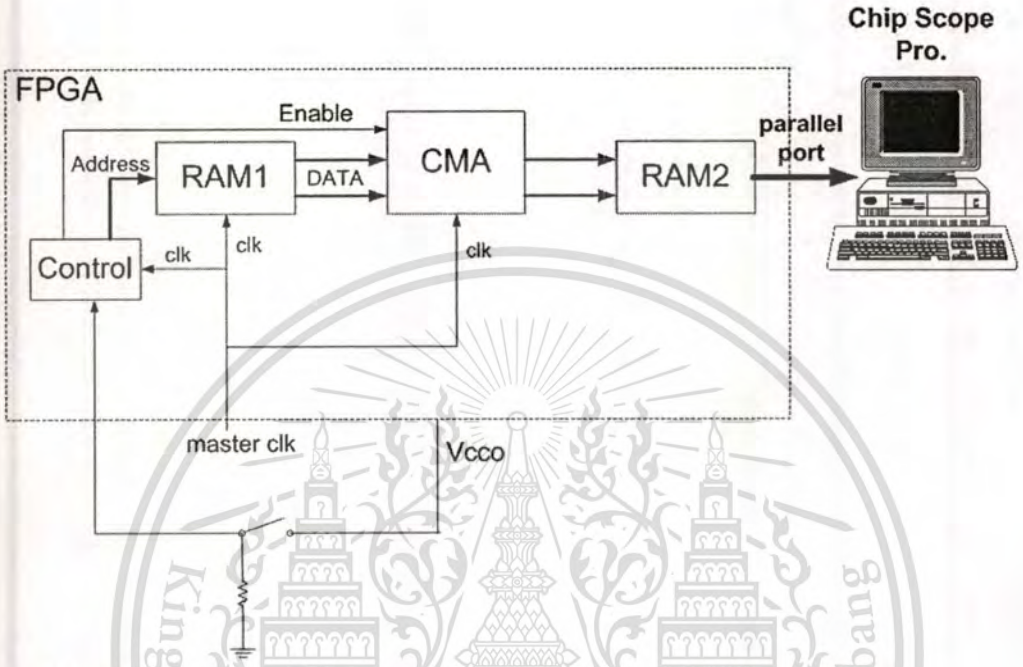


Fig. 6.3 Testing system of the adaptive processing unit with the proposed DCMA-based pipelined architecture.

6.4.2. Fixed-Point Arithmetic

In this section, the adaptive array processing unit with the proposed DCMA-based pipelined architecture was tested. Fixed-point operations were introduced to reduce either the power consumption or resource for FPGA implementation. Fig. 6.4-6.5 show the trajectories of real and imaginary parts of weights obtained from the test and simulation of the proposed unit using fixed-point arithmetic. As seen in the figures, the testing results have only 2048 iterations while the simulation results have 8,000 iterations. This is because of the limited resource for implementing RAMs. The figures reveal that weight trajectories obtained from testing have in fact a good agreement with that obtained from simulation. These imply that the implemented unit can operate correctly. The final values of weights (at 2,047 iterations) obtained from the simulation and testing were used to calculate the final antenna pattern as seen in Fig. 6.6. The figure shows that the patterns obtained from the simulation and testing

are almost identical and have nullity in the direction of 121° close to the incident angle of the interference signal (ϕ_I = 120°). The SINRs from both simulation and test were -27.2 dB. At 2,047 iterations, the DCMA adaptive antenna does not completely converge. The SINRs will increase if the implemented unit is test at more iterations.

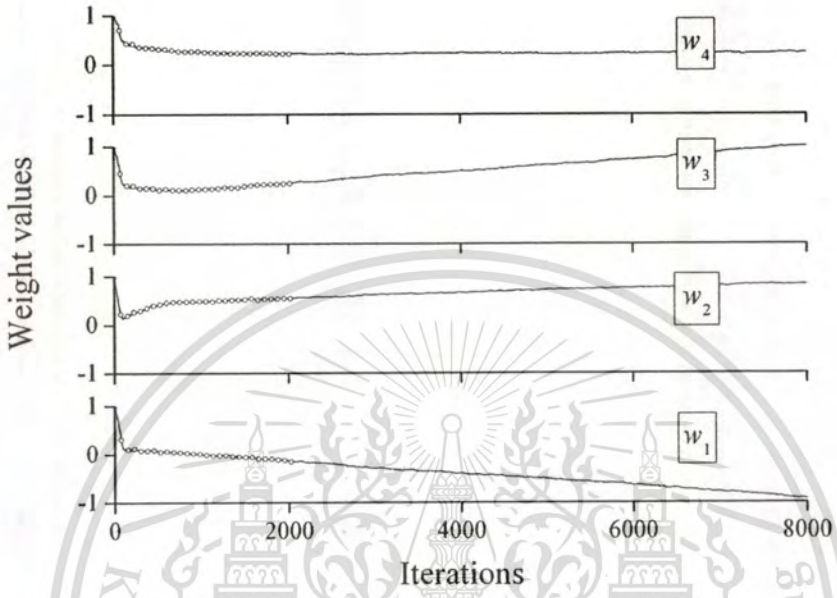


Fig. 6.4 Trajectories of real part of weights (– and o denote simulation and test, respectively).

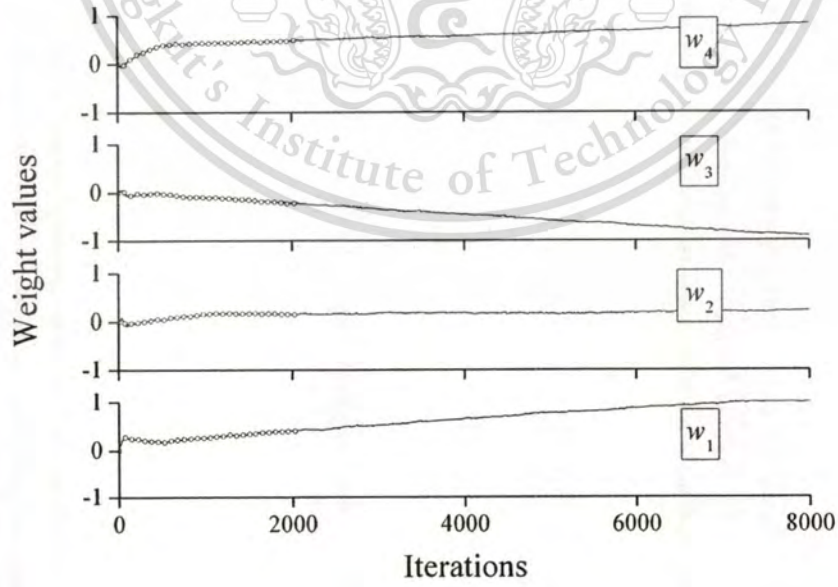


Fig. 6.5 Trajectories of imaginary part of weights (– and o denote simulation and test, respectively).

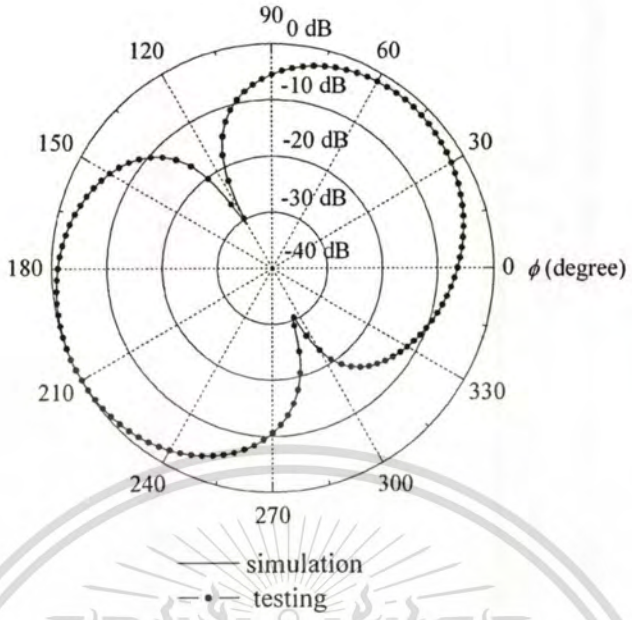


Fig. 6.6 Antenna pattern obtained from the proposed adaptive antenna with the proposed unit using fixed-point operations (at 2,047 iterations).

The output of the proposed unit obtained from the simulation and testing should be considered that its eye-pattern after convergence must be opened. Fig. 6.7-6.8 illustrates the output constellation of the adaptive array processing unit with the DCMA-based pipelined architecture using fixed-point operations. The output constellations obtained from the simulation and testing of the unit between 0-2,047 iterations are shown in Fig. 6.7 (a) and (b), respectively. Some of the output signals are not completely concentrated as QPSK signals while some have scattering similar QPSK signals. This obviously reveals that the adaptive array processing unit attempts to eliminate the interference signal, but the adaptation is not complete, corresponding to its weight trajectories. Fig. 6.8 (a) and (b) show the output constellation obtained from simulation and test between 1,001-2,047 iterations, respectively. In the figure, both eye-patens are almost completely opened. Between 1,001-2,047 iterations, scattering indicates that convergence of the antenna is almost stable.

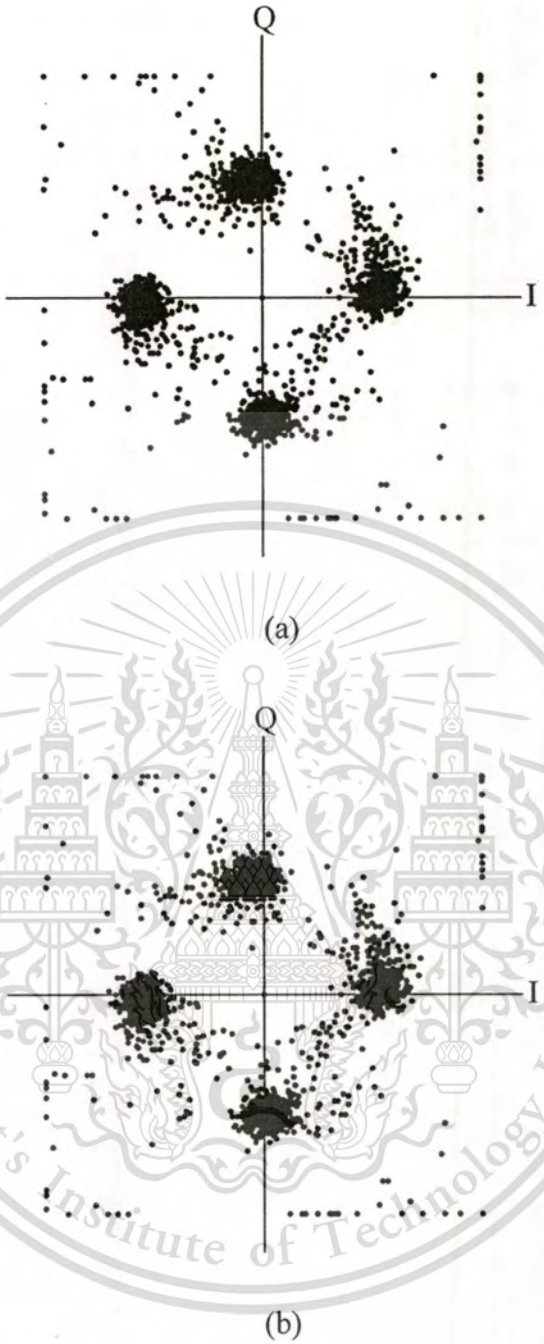
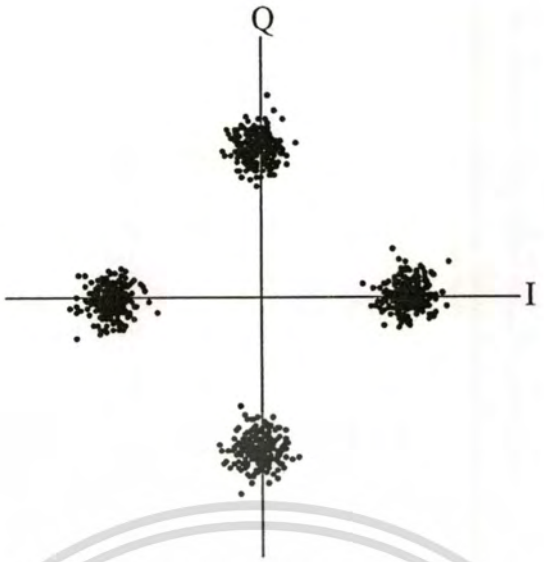
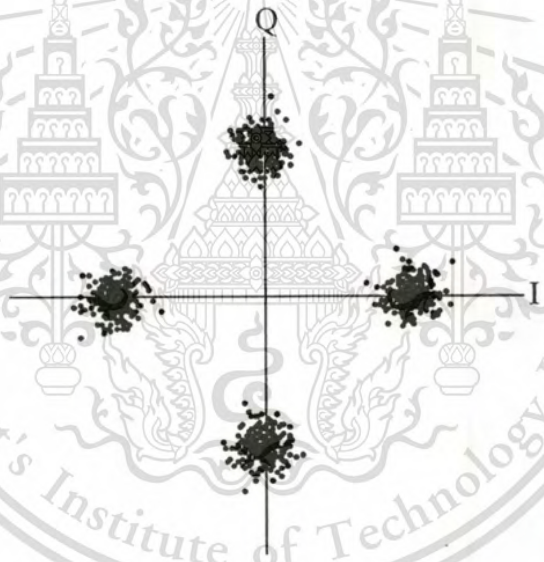


Fig. 6.7 Output constellation of the proposed unit using fixed-point operators between 0-2,047 iterations obtained from (a) simulation (b) testing of FPGA.



(a)



(b)

Fig. 6.8 Output constellation of the proposed unit using fixed-point operators between 1,001-2,047 iterations obtained from (a) simulation (b) testing of FPGA.

6.4.3. Power-of-two Arithmetic

The adaptive array processing unit with the proposed DCMA-based pipelined architecture using power-of-two operations was implemented and then tested in this section. The fixed-point multipliers as presented in Chapter 4 for multiplication

between input signals $\mathbf{x}(k)$ and others were replaced by the power-of-two multipliers. Fig. 6.9-6.10 show the trajectories of real and imaginary parts of weights obtained from simulation and test of the unit. Because of the limited resource for implementing RAMs, the testing results have 2,048 samples while the simulation results have 8,000 samples. In the figure, weights obtain from test of FPGA agree with that obtained from simulation. The antenna pattern obtained from multiplication of a weight vector $\mathbf{w}(k)$ and array response matrix $\mathbf{H}$ is illustrated in Fig. 6.11 in which k = 2,047. The nullity of the antenna appears at 127° and main beam directs to 37°. The SINRs which are calculated from the antenna patterns obtained from the simulation and testing were 18.4 dB. At k = 2,047, the adaptive antenna doest not converge as seen in weight trajectories in Fig. 6.9-6.10. Comparing with the unit using fixed-point operations, the convergence rate of the adaptive antenna with the unit using power-of-two operations is faster.

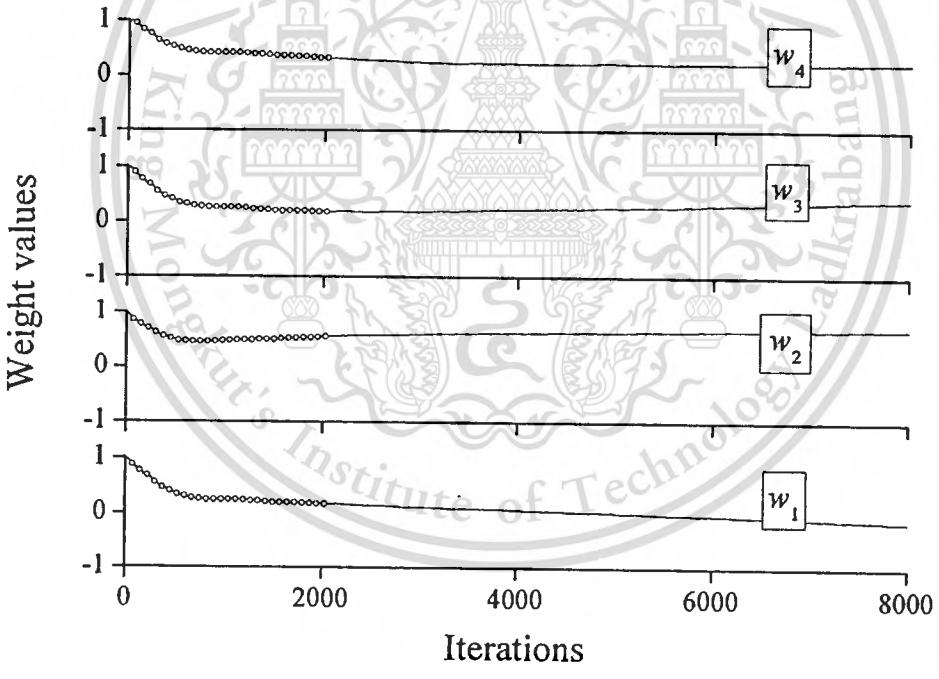


Fig. 6.9 Trajectories of real part of weights (– and o denote simulation and test, respectively).

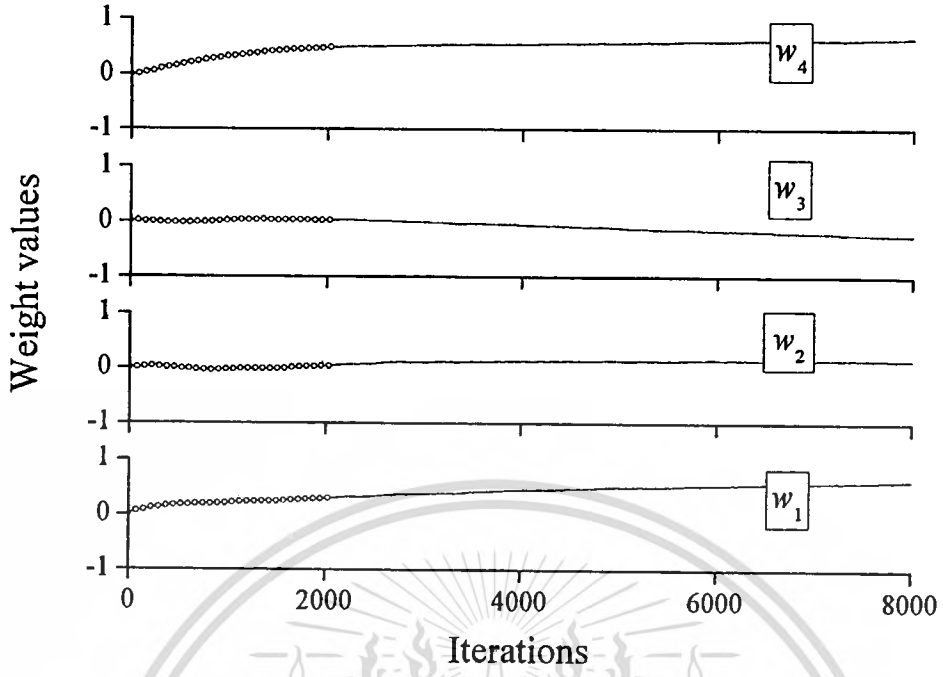


Fig. 6.10 Trajectories of real part of weights (– and o denote simulation and test, respectively).

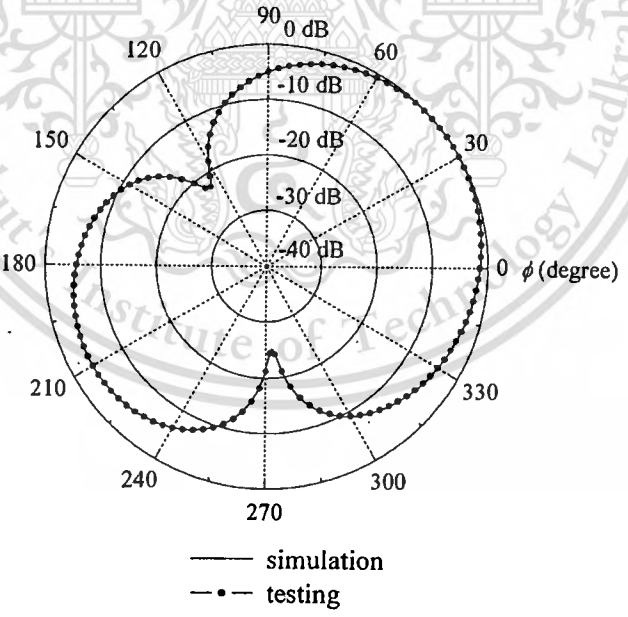
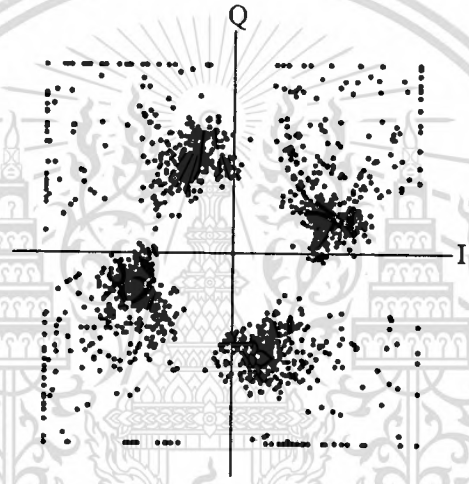
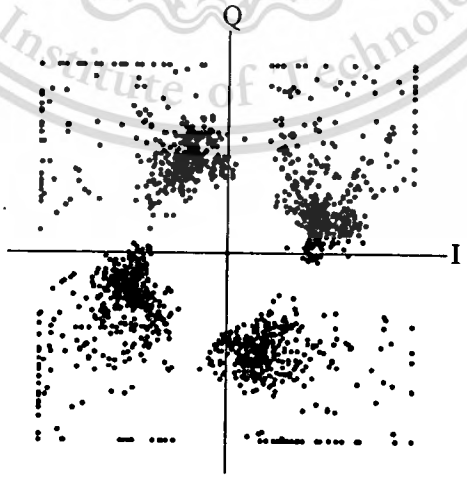


Fig. 6.11 Antenna pattern obtained from the proposed adaptive antenna with the proposed unit using power-of-two operations (at 2,047 iterations).

The output of the DCMA adaptive array processing unit using power-of-two operations obtained from the simulation and testing was considered. Fig. 6.12 illustrates the output constellation of the proposed unit with the proposed architecture using power-of-two operators between 0-2,047 iterations. Note that the eye-pattern obtained from simulation as seen in Fig. 6.12 (a) is almost identical to that obtained from testing as seen in Fig. 6.12 (b). These eye-patterns are closed. The output constellation of the proposed unit using power-of-two operators between 1,001-2,047 iterations is shown in Fig. 6.13 in which it implies that the eye-pattern is opened. It is also noticed that the output constellation obtained from test is almost identical to that obtained from the test.



(a)



(b)

Fig. 6.12 Output constellation of the proposed unit using power-of-two operators between 0-2,047 iterations obtained from (a) simulation (b) testing of FPGA.

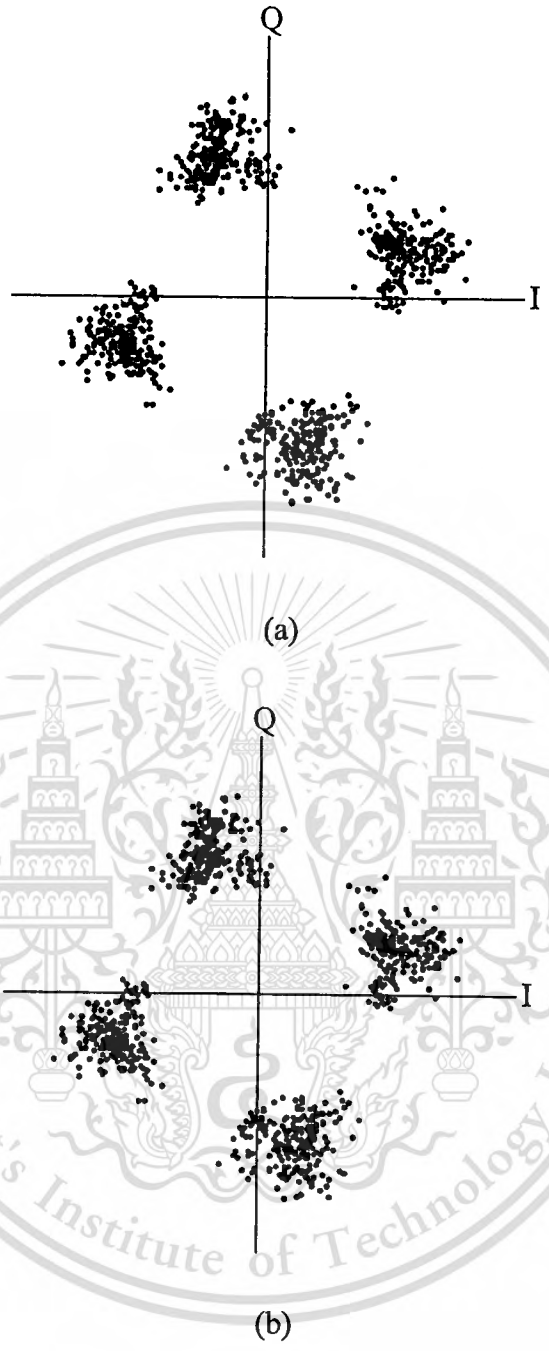


Fig. 6.13 Output constellation of the proposed unit using power-of-two operators between 1,001-2,047 iterations obtained from (a) simulation (b) testing of FPGA.

6.4.4. Resource and Cost of FPGA

The pipelined 5D CMA was implemented on the Xilinx Virtex-E XCV400E FPGA device. Based on the design by using fixed-point operations, the maximum operating clock frequency of FPGA was 69.1 MHz. In total, the equivalent gates of 63,614 were used. By replacing fixed-point multipliers by power-of-two multipliers,

the maximum operating clock frequency of FPGA decreases to 65.2 MHz. The total equivalent gates of 46,514 were used. Note that after replacing the multipliers, it can reduce 26.9 percent of the use of resource for FPGA implementation. The percentage of the reduced resource will increase when the more accuracy of CMA is required. The maximum clock operating frequency of FPGA designed by using power-of-two multipliers slightly reduces. This is because its critical path which is about $3T_a$ or T_m increases. Generally, T_a and T_m depend on the wordlength of either an adder or a multiplier, respectively. The critical path limited by 3 adders or 1 multiplier proportionally increase with their wordlenghts. In this thesis, by replacing fixed-point multipliers by power-of-two multipliers, the wordlenght of weights for adaptation is increased from 10 bits to be 12. Thus, the critical path in whole architecture of DCMA will increase. This problem of increasing the critical path can be resolved by increasing the number of delay units.

6.5 Summary

The adaptive array processing unit implemented on FPGA with the proposed DCMA-based pipelined architecture was tested. The testing results show that the implemented unit using fixed-point and power-of-two operations can operate accurately in parallel. The implemented unit using fixed-point and power-of-two operations utilizes the total equivalent gates of 63,614 and 46,514, respectively. This reveals that replacing fixed-point multipliers by power-of-two multipliers can significantly reduce the use of resource for implementation of FPGA with the percentage of 26.9. The unit implemented in FPGA using fixed-point and power-of-two operations can be run at the maximum clock frequency of 69.1 and 65.2 MHz, respectively. This is sufficient for operation in the IMT2000 communication system.

CHAPTER 7
CONCLUSIONS AND DISCUSSIONS

Since the demand of the service from mobile communication systems has decreased rapidly, enlarging the system bandwidth is an alternative to make the requirement possible. However, the electromagnetic spectrum which is a valuable resource is limited. Thus, the direct method to increase the system capacity by opening the new spectrum space is not feasible in the future.

Another way to increase the capacity of mobile communication system is to reduce interference effect in the systems. Recently, the adaptive antenna has been received considerable interests to resolve the underlying problem. Therefore, one of the goals of this thesis is to present an adaptive array antenna which can reduce interference. The adaptive array antenna employs CMA to adjust its weights for adaptation because it is easy to implement CMA. However, an important drawback of applying CMA is its slow convergence.

In chapter 3, the low-cost initialization technique for CMA is presented to improve the convergence property of CMA. The technique exploits a switched beam of the flat four-beam phased array antenna with maximum received power to initial CMA. There are four main beams in different directions namely 45°, 135°, 225°, and 315°. The simulation results have shown that the proposed technique can significantly improve the convergence property of CMA. In addition, the proposed adaptive antenna system was implemented and then experimented to confirm the feasibility of the proposed low-cost hardware-assisted initialization technique.

In chapter 4, the new antenna, a switched-beam element phased array antenna, replaces the proposed previous one. It can generate more possible initial beams for CMA by four one-bit phase shifters and applying forward or reverse bias to PIN diodes. Twelve initial beams for CMA were selected to cover the service area of 360°. The directions of those initial main beams are in 21°, 45°, 69°, 111°, 135°, 159°, 201°, 225°, 249°, 291°, 315°, and 339°. The proposed adaptive antenna system with the new antenna has the similar process to that with the previous one to select an initial beam with maximum received power. The simulation results showed that the proposed adaptive antenna system with the switched-beam element phased array antenna has the fastest convergence rate compared to the convergence rate obtained from other systems, i.e. the system with the

flat four-beam phased array antenna and the system without beam-switching initialization.

Although the proposed adaptive antenna system can resolve the main problem of the slow convergence of CMA, the implemented adaptive processor with MAC architecture cannot operate at the high-frequency clock. Another goal of this thesis is to propose the architecture of the CMA adaptive processing unit which can operate at high-frequency clock. The efficient architecture of the CMA array processing unit was proposed in Chapter 5. The technique utilizes the ability of pipelining to increase the speed performance of FPGA which is a target device. The unit runs in parallel. Practically, fixed-point operations were used to implement the unit. In this thesis, fixed-point multipliers were also replaced by power-of-two multipliers for multiplication between input signals and other to reduce the use of resource of FPGA implementation. The 10 and 12-bit weights of fixed-point and power-of-two operations were chosen, respectively in accordance with the resulting SINR above 20 dB. After replacing fixed-point multipliers by power-of-two multipliers, the use of resource for FPGA implementation can be reduced with the percentage of 26.9 as shown in Chapter 6. The results obtained from testing implemented FPGA have a good agreement with that obtained from simulations. The testing results reveal that the implemented FPGA can eliminate interference. The maximum operating clock frequencies of the implemented FPGA with fixed-point operations and power-of-two operations are 69.1 MHz 65.2 MHz sufficient for operation in the high-speed communication system. Note that with the proposed pipelined architecture, the implemented CMA array processing unit can operate in real time.

Althought one-bit phase shifters can offer to as a cost effective solution to switch the radiation pattern of an antenna in order to improve the convergence property of CMA, some additional associated hardwares i.e. RF detector , PIN diodes are required. With similar concept to the proposed technique, digital beam synthesis should be employed instead of the RF phase shifters to form beams in digital domain. The beams obtained from the technique of digital beam synthesis can be used to initial CMA. The initial beam should be selected through a digital power detector achieved by FPGA. Both digital beam synthesis and power detector can be implemented on only FPGA, Thus, this can be referred to as a compact system by omitting RF phase shifters and RF power detector.

REFERENCES

[1] A. Anderson, M. Millnert, M. Viberg, and B. Wahlberg, "An Adaptive Array for Mobile Communications Systems," *IEEE Trans. Veh. Technol.*, vol. 33, pp. 144-155, Aug. 1984.

[2] H. V. Poor and G. W. Wornell, *Wireless Communications Signal Processing Perspectives*, Prentice-Hall, New Jersey, 1998.

[3] S. Haykin (Eds.), *Unsupervised Adaptive Filtering*, New York, John Willey & Sons, 2000.

[4] I.C. Liberti, Jr. and T. S. Rappaport, *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, New Jersey, Prentice-Hall, 1999.

[5] D.N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Commun.*, vol. COM-28(11), pp. 1867-1875, Nov. 1980.

[6] R. Gooch and J. Lundell, "The CM array: An adaptive beamformer for constant modulus signal," *Proc. of ICASSP'86*, vol. 4, pp. 2523-2526, Apr. 1986.

[7] J. R. Treichler and M. G. Larimore, "New Processing techniques based on the constant modulus adaptive algorithm," *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-33, pp. 420-431, Apr. 1985.

[8] M. Fujimoto, N. Kikuma, and N. Inagaki, "Performance of CMA adaptive array optimized by Marquardt method for suppressing mutipath wave," *IEICE Trans.*, vol. J74-B-II, no. 11, pp. 599-607, Nov. 1991.

[9] H. Furukawa, Y. Kamio, and H. Sasaoka, "Co-channel interference reduction method using CMA adaptive array antenna," *Proc. of IEEE PIMRC'96*, Taipei, pp. 512-516, Oct. 1996.

[10] T. Ohgane, T. Shimura, N. Matsuzawa ,and H. Sasaoka, "An Implementation of a CMA Adaptive Array for High Speed GMSK Transmission in Mobile Communications," *IEEE Trans. Veh. Technol.*, vol. 42, no. 3, pp. 282-288, Aug. 1993.

[11] T. Ohgane, N. Matsuzawa, T. Shimura, M. Mizuno, and H. Sasaoka, "BER Performance of CMA Adaptive Array for High-Speed GMSK Mobile Communication—A Description of Measurements in Central Tokyo," *IEEE*

Trans. Veh. Technol., vol. 42, no. 4, pp.484-490, Nov. 1993.

[12] J. Litva and T. K. Lo, *Digital Beamforming in Wireless Communications*, John Walker, USA:, 1996.

[13] I Chiba, R Miura, T. Tanaka and Y. Karasawa, "Digital Beam Forming (DBF) Antenna System for Mobile Communications," *IEEE AES Systems Magazine*, pp. 31-41, Sept. 1997.

[14] H. Furukawa, Y. Kamio, and H. Sasaoka, "Co-channel interference canceller using CMA adaptive array antenna," *Electronics Letters*, vol. 33, no. 13, pp.1106-1108, June 1997.

[15] H. Furukawa, Y. Kamio, and H. Sasaoka, "Cochannel Interference Reduction and Path-Diversity Reception Technique Using CMA Adaptive Array Antenna in Digital Land Mobile Communications," *IEEE Trans. Veh. Technol.*, vol. 50, no. 2, pp. 605-615, Mar. 2001.

[16] S. Denno and T. Ohira, "Modified Constant Modulus Algorithm for Digital Signal Processing Adaptive Antennas With Microwave Analog Beamforming," *IEEE Trans. Antennas and Propagation*, vol. 50, no. 6, pp. 850-857, June 2002.

[17] A. Mathur, A. V. Keerthi, and J. J. Shynk, "A Variable Step-Size CM Array Algorithm for Fast Fading Channels," *IEEE Trans. Signal Processing*, vol. 45, no. 4, pp. 1083-1087, Apr. 1997.

[18] M. G. Larimore and J. R. Treichler, "Convergence Behavior of the Constant Modulus Algorithm," *Proc. of IEEE ICASSP 83*, Boston, pp. 13-16, 1983.

[19] N. Kimuma and K. Takao, "Effect of initial values of adaptive array," *IEEE Trans. Aerospace and Electronic Systems*, vol. AES-22, no. 6, pp. 688-694, Nov. 1986.

[20] K. Takao and H. Matsuda, "The choice of the initial condition of CMA adaptive arrays," *IEICE Trans. Commun.*, vol. E78-B, no. 11, pp. 1474-1479, Nov. 1995.

[21] R. Yonezawa and I. Chiba, "A Combination of Two Adaptive Algorithms SMI and CMA," *IEICE Trans. Commun.*, vol.E84-B, no.7, pp. 1768-1773, July 2001.

[22] Y. Chen, T. Le-Ngoe, B. Champagne, and C. Xu, "Recursive Least Squares Constant Modulus Algorithm for Bind Adaptive Array," *IEEE Trans. Signal Processing*, vol. 52, no. 5, pp. 1452-1456, May 2004.

[23] T. Tanaka, R. Miura, I. Chiba, and Y. Karasawa, "An ASIC Implementation Scheme to Realize a Beam Space CMA Adaptive Array Antenna," *IEICE*

Tans. Commun., vol. E78-B, no. 11, Nov. 1995.

[24] K. K. Parhi, "Algorithm transformation techniques for concurrent processors," *Proc. of the IEEE*, vol. 77, Issue 12, pp. 1879 – 1895, Dec. 1989.

[25] K. K. Parhi and D. G. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters. I. Pipelining using scattered look-ahead and decomposition," *IEEE Trans. on Acoustics, Speech, and Signal Proc.*, vol. 37, pp. 1099 – 1117, July 1989.

[26] K. K. Parhi and D. G. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters. II. Pipelined incremental block filtering," *IEEE Trans. on Acoustics, Speech, and Signal Proc.*, vol. 27, pp. 1118–1134, July 1989.

[27] G. Long, F. Ling, and J. G. Proakis, "The LMS algorithm with delayed coefficient adaptation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, pp. 1397–1405, Sep. 1989.

[28] M. D. Meyer and D. P. Agrawal, "A High Sampling Rate Delayed LMS Filter Architecture," *IEEE Trans. Circuits and Syst. II*, vol. 40, no. 11, pp. 727-729, Nov. 1993.

[29] N. R. Shanbhag and K. K. Parhi, "Relaxed look-ahead pipelined LMS adaptive filters and their application to ADPCM coder," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 753–766, Dec. 1993.

[30] K. Matsubara, K. Nishikawa, and H. Kiya, "Pipelined LMS Adaptive Filter Using a New Look-Ahead Transformation," *IEEE Trans. Circuits and Syst. II*, vol. 46, no. 1, pp. 51-55, Jan. 1999.

[31] L. Ting, R. Woods, and C. F. N. Cowan, "Virtex FPGA Implementation of a Pipelined Adaptive LMS Predictor for Electronic Support Measures Receivers," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 13, No. 1, Jan. 2005.

[32] N. Benvenuto, L. E. Franks, and F.S. Hill, JR., "On the Design of FIR Filters with Power-of-two Coefficients," *IEEE Trans. Commun.*, vol. com-32, no. 12, pp. 1299-1307, Dec. 1984.

[33] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete powers-of-two coefficient space," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, pp. 583-591, June 1983.

[34] Q. Zhao and Y. Tadokoro, "A simple design of FIR filters with powers- of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 566-570, May 1988.

[35] W. J. Oh and Y. H. Lee, "Implementation of Programmable Multiplierless FIR Filters with Powers-of-two Coefficients," *IEEE Trans. Circuits Syst.*, vol. 42, no. 8, pp. 553-556, Aug. 1995.

[36] P. Xue and B. Liu, "Adaptive equalizer using finite-bit power-of-two quantizer," *IEEE Trans. Acoust., Speech, and Signal Processing,* vol. asp-34, no. 6, pp. 1603-1611, Dec. 1986.

[37] M. Krairiksh, P. Ngamjanyaporn and C. Kessuwan, "A Flat Four-Beam Compact Phased Array Antenna," *IEEE Microwave And Wireless Components Letters*, vol. 12, no 5, pp. 184 – 186, May 2002.

[38] A. Boonpoonga, P. Sirisuk and M. Krairiksh, "Improvement of CMA Adaptive Phased Array Antenna by using Swithced-Beam Initialization", *Proc. of the 2005 International Technical Conference on Circuits/System, Computers and Communications (ITC-CSCC 2005)*, Korea, pp. 221-222, July 2005.

[39] C. A. Balanis, *Antenna Theory: Analysis and Design.* 2nd ed. New York: John Wiley & Sons, 1997.

[40] I. Fijalkow,A. Touzni, and J.R. Treichler, "Fractionally spaced equalization using CMA: robustness to channel noise and lack of disparity," *IEEE Trans. Signal Processing*, vol. 45, No. 1, Jan. 1997, pp. 56 – 66.

[41] Y. Li and Z. Ding, "Convergence analysis of finite length blind adaptive equalizers," *IEEE Trans. Signal Processing,* vol. 43, No. 9, Sep. 1995, pp. 2120 – 2129.

[42] J. R. Treichler and B. G. Agee "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 31, No. 2, Apr. 1983, pp. 459- 472.

[43] G. J. Foschini, "Equalizing without altering or detecting data," *AT & T technical journal*, vol. 64, No. 88, 1985, pp. 1885-1911.

[44] D. Liu and L. Tong, "An Analysis of Constant Modulus Algorithm for Array Signal Processing," *Signal Processing*, vol. 73, 1999, pp. 81-104.

[45] J.E. Hudson, *Adaptive Array Principle. IEE Electromagnetic Waves Seiress 11*, London: Perter Peregrinus Ltd., 1991.

[46] Virtex™-E 1.8 V Field Programmable Gate Arrays, *Xilinx product specification*, Jan. 16, 2006.

[47] P. Fletcher and M. Dean, "Low complexity implementation of LMS algorithm," *Electronic letters*, vol. 38, no. 15, July 2002, pp. 836 - 837.

[48] R. Prasad, W. Mohr and W. Konhauser, *Third Generation Mobile Communication Systems*, Boston : Artech House, 2000.

[49] P. Ngamjanyaporn, *A Phased Array antenna of Switched-beam Elements*, Doctoral thesis, School of graduate studies, King mongkut's Institute of Technology Ladkrabang, 2005.

[50] Y. Ogawa, M. Ohmiya, K. Itoh, "Behaviors of an LMS adaptive array for multipath fading reduction," *IEICE Letter Antenna and propagation*, vol. E 67, July 1984, p. 395, 396.

[51] D. A. Patternson and J. L. Hennessy, *Computer Organization & Design the Hardware/ Software Interface*, Morgan Kaufmann 2's edition, 1997.

[52] A. G. Derneryd, "Analysis of the Microstrip Disk Antenna Element," *IEEE Trans. Antennas and Propagation*, vol. AP-27, no. 5, Sept. 1979, pp. 660-664.

[53] H. Yuan, K. Hirasawa, and Y. Zhang "The Mutual Coupling and Diffraction Effects on the Performance of a CMA Adaptive Array," *IEEE Trans. on Vehicular Technology*, vol. 47, No. 3, Aug. 1998, pp. 728-736.

[54] P. Ngamjanyaporn and M. Krairiksh: "Switched-beam single patch antenna," *Electronic Letters*, Vol. 38, NO. 1, pp. 7-8, Jan. 2002.

[55] P. Ngamjanyaporn, C. Phongcharoenpanich, P. Akkaraekthalin and M. Krairiksh: "Signal-to-Interference Ratio Improvement by Using a Phased Array Antenna of Switched-Beam Elements," *IEEE Trans. Antennas and Propagation*, Vol. 53, No. 5, pp.1819-1828, May 2005.

[56] A. Boonpoonga, P. Sirisuk and M. Krairiksh, "A Switched-beam Element Phased Array Antenna for CMA Initialization," *Proc. of the 36th European Microwave Conference (EuMC)*, Manchester, pp.772-775, Sept. 2006.

[57] A. Boonpoonga, P. Sirisuk, C. Phongcharoenpanich, and M. Krairiksh, "Beam-switching-assisted CMA initialization for Handset Adaptive Antenna," *Proc. of the 2007 IEEE Radio & Wireless Symposium (RWS)*, California, pp. 479-482, Jan. 2007.

[58] A. Boonpoonga, P. Sirisuk, M. Chongcheawchamnan, S. Patisang, and M. Krairiksh, "Low-Cost Hardware-Assisted Initialization for CMA Adaptive Antenna," IET *Microwaves, Antennas and Propagation Trans.* (in Press).

[59] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*, New Jersey, Pentice Hall, 1995.

[60] S. J. Piestrak, "Design of Squarers Modulo A With Low-Level Pipelining," *IEEE*

Trans. Circuits and Syst. II, vol. 49, no. 1, pp. 31-41, Jan. 2002.

[61] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*, John Wiley & Sons, Inc., Canada, 1999.

RELATED PUBLICATIONS

1). A. Boonpoonga, P. Sirisuk and M. Krairiksh, "Improvement of CMA Adaptive Phased Array Antenna by using Swithced-Beam Initialization," *Proc. of the 2005 International Technical Conference on Circuits/System, Computers and Communications (ITC-CSCC 2005)*, Korea, pp. 221-222, July 2005.

2). A. Boonpoonga, P. Sirisuk and M. Krairiksh, "Wordlength Effect on CMA Adaptive Phased Array Antenna Using Fixed-point Arithmetic," *Proc. of the 2005 Asia-Pacific Microwave Conference (APMC2005)*, Suzhou, China, vol. 5, pp. 3024-3027, Dec. 2005.

3). A. Boonpoonga, P. Sirisuk and M. Krairiksh, "A Switched-beam Element Phased Array Antenna for CMA Initialization," *Proc. of the 36th European Microwave Conference (EuMC)*, Manchester, pp.772-775, Sept. 2006.

4). A. Boonpoonga, P. Sirisuk,C. Phongcharoenpanich, and M. Krairiksh, "Beam-switching-assisted CMA initialization for Handset Adaptive Antenna," *Proc. of the 2007 IEEE Radio & Wireless Symposium (RWS)*, Californai, pp. 479-482, Jan. 2007 .

5). A. Boonpoonga, P. Sirisuk,C. Phongcharoenpanich, and M. Krairiksh, "Performance of Switched-beam Element Phased Array for Handset Adaptive Antenna," *Proc. of the 2007 European Conference on Wireless Technologies (ECWT)*,Munich, pp. 209-212, Oct. 2007.

6). A. Boonpoonga, S. Buritramart, P. Sirisuk, M. Krairiksh, T. K. Sarkar, "Adaptive Array Processing Unit Based on Direct Data Domain Least Square Approach Using Conjugate Gradient Method," *Proc. of the 2007 Asia-Pacific Microwave Conference (APMC2007)*, Bangkok, pp. 171-174, Dec. 2007.

7). A. Boonpoonga, P. Sirisuk, M. Chongcheawchamnan, S. Patisang, and M. Krairiksh, "Hardware-Assisted Initialization for CMA Adaptive Antenna," *IET Microwaves, Antennas and Propagation Trans.* (in Press).

AUTHOR BIOGRAPHY

Akkarat Boonpoonga was born on April 2, 1980 in Surin, Thailand. He received the B. Eng. degree in Electrical Engineering from King Mongkut's Institute of Technology NorthBangkok, Bangkok, Thailand, in 2002 and the M.Eng. degree in Telecommunications Engineering from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 2004. During 2004-2007, the Thailand Research Fund (TRF) supported his study through The Royal Golden Jubilee (RGJ) program. He was with Prof. Tapan K. Sarkar in Syracuse University as a visiting scholar during February-June, 2007. His research interests include an adaptive array antenna and FPGA design and implementation.