



1

**AN APPLICATION OF THE CEREBELLAR MODEL  
ARTICULATION CONTROLLER FOR A SWITCHED  
RELUCTANCE ROTOR POSITION ESTIMATOR**



**Jenifer M. Shannon  
Air Vehicle and Crew Systems Technology Department  
NAVAL AIR WARFARE CENTER, AIRCRAFT DIVISION  
Warminster, PA 18974**

**S DTIC  
ELECTE  
NOV 01 1993  
A D**

**NOVEMBER 1992**

**FINAL REPORT  
Work Unit Plan No. IRA11  
Project ID No. 01B1201**

*Approved for public release: distribution is unlimited*

**93-26258**



93 10 23 037

## NOTICES

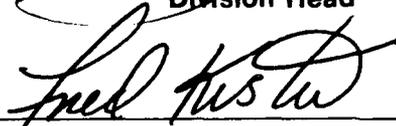
**REPORT NUMBERING SYSTEM** — The numbering of technical project reports issued by the Naval Air Warfare Center, Aircraft Division, Warminster is arranged for specific identification purposes. Each number consists of the Center acronym, the calendar year in which the number was assigned, the sequence number of the report within the specific calendar year, and the official 2-digit correspondence code of the Functional Department responsible for the report. For example: Report No. NAWCADWAR-92001-60 indicates the first Center report for the year 1992 and prepared by the Air Vehicle and Crew Systems Technology Department. The numerical codes are as follows:

CODE	OFFICE OR DEPARTMENT
00	Commanding Officer, NAWCADWAR
01	Technical Director, NAWCADWAR
05	Computer Department
10	AntiSubmarine Warfare Systems Department
20	Tactical Air Systems Department
30	Warfare Systems Analysis Department
50	Mission Avionics Technology Department
60	Air Vehicle & Crew Systems Technology Department
70	Systems & Software Technology Department
80	Engineering Support Group
90	Test & Evaluation Group

**PRODUCT ENDORSEMENT** — The discussion or instructions concerning commercial products herein do not constitute an endorsement by the Government nor do they convey or imply the license or right to use such products.

Reviewed By:  Date: 9/1/93  
Branch Head

Reviewed By:  Date: 9/1/93  
Division Head

Reviewed By:  Date: 9/3/93  
Director/Deputy Director

## REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified		1b RESTRICTIVE MARKINGS	
2a SECURITY CLASSIFICATION AUTHORITY		3 DISTRIBUTION AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited	
2b DECLASSIFICATION/DOWNGRADING SCHEDULE			
4 PERFORMING ORGANIZATION REPORT NUMBER(S) NAWCADWAR-93041-60		5 MONITORING ORGANIZATION REPORT NUMBER(S)	
6a NAME OF PERFORMING ORGANIZATION Air Vehicle and Crew Systems Technology Department	6b OFFICE SYMBOL (If applicable) 6012	7a NAME OF MONITORING ORGANIZATION	
6c ADDRESS (City, State, and ZIP Code) Naval Air Warfare Center Aircraft Division Warminster, PA 18974-5000		7b ADDRESS (City, State, and ZIP Code)	
8a NAME OF FUNDING/SPONSORING ORGANIZATION NAVAIRWARCENACDIWAR	8b OFFICE SYMBOL (If applicable) 01B	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c ADDRESS (City, State, and ZIP Code)		10 SOURCE OF FUNDING NUMBERS	
		PROGRAM ELEMENT NO.	PROJECT NO 01B1201
		TASK NO	WORK UNIT ACCESSION NO IR11
11. TITLE (Include Security Classification) An Application of the Cerebellar Model Articulation Controller for a Switched Reluctance Rotor Position Estimator (Unclassified)			
12. PERSONAL AUTHOR(S) Jenifer M. Shannon			
13a. TYPE OF REPORT Final	13b TIME COVERED FROM 2/92 TO 11/92	14 DATE OF REPORT (Year, Month, Day) 1992 November	15 PAGE COUNT 53
16. SUPPLEMENTARY NOTATION			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP	
19. ABSTRACT (Continue on reverse if necessary and identify by block number) A method of estimating the rotor position of a switched reluctance machine without the need for a rotor-mounted position sensor has been developed. This method takes advantage of the information derived from known phase voltage and current waveforms. The information is fed as the inputs to a neural network, which after being trained, can correctly map the rotor position to its output. The most accurate mapping results were obtained using a Cerebellar Model Articulation Controller (CMAC) neural network. The performance of the neural network has been tested with measured waveforms from a three phase 120 HP switched reluctance motor. It successfully maps the rotor position with an average root mean square error of one tenth of a mechanical degree.			
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL Jenifer M. Shannon		22b. TELEPHONE (Include Area Code) 215-441-7216	22c. OFFICE SYMBOL 6012

TABLE OF CONTENTS

Abstract ..... ii

List of Figures ..... iv

List of Tables ..... iv

1.0 Introduction ..... 1

    1.1 Background: The More Electric Aircraft Concept ..... 1

    1.2 The Application: Switched Reluctance Machines ..... 2

        1.2.1 Machine Design & Operation ..... 2

        1.2.2 Advantages of Switched Reluctance ..... 5

        1.2.3 Critical Design Issues ..... 5

2.0 Method ..... 8

    2.1 The Paradigm: CMAC ..... 8

        2.1.1 A Historical Perspective ..... 8

        2.1.2 The Benefits of CMAC ..... 9

        2.1.3 The Algorithm ..... 10

            2.1.3.1 Input Quantization ..... 10

            2.1.3.2 Virtual Address Computation ..... 11

            2.1.3.3 Hash Coding Procedure ..... 12

            2.1.3.4 Output Computation ..... 13

        2.1.4 Hardware Implementation ..... 13

        2.1.5 Summary ..... 13

    2.2 Data Preprocessing ..... 14

    2.3 The Importance of CMAC Parameters ..... 21

3.0 Results ..... 23

    3.1 Comparison with Other Paradigms ..... 23

    3.2 Input Redundancy ..... 25

    3.3 Optimized CMAC Network ..... 26

    3.4 Discussion ..... 28

    3.5 Conclusions ..... 29

References ..... 30

Appendix A: Glossary ..... 31

Appendix B: Filtered Inputs Used for Training ..... 36

UNCLASSIFIED//FOR OFFICIAL USE ONLY

Accession For	
NTIS - CRA-1	<input checked="" type="checkbox"/>
DTIC - TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution _____	
Availability Office _____	
Dist	
A-1	

**LIST OF FIGURES**

Figure 1.2.1.a Cross Sectional View of a SRM.....	3
Figure 1.2.1.b Converter Circuitry .....	3
Figure 1.2.1.c Machine Operation .....	4
Figure 1.2.3.a Magnetic Model Approach .....	6
Figure 1.2.3.b Neural Network Approach .....	7
Figure 2.1.3 CMAC Mapping Scheme.....	10
Figure 2.2.a Data Columns versus Record Number.....	16
Figure 2.2.b Reordered Data Columns versus Record Number .....	17
Figure 2.2.c Stripped Data Columns versus Record Number .....	18
Figure 2.2.d Normalized and Filtered Data versus Record Number .....	20
Figure 3.1.a Backpropagation Recall Capability .....	23
Figure 3.1.b Radial Basis Function Recall Capability .....	24
Figure 3.1.c CMAC Recall Capability .....	25
Figure 3.3.a CMAC Network Output .....	27
Figure 3.3.b CMAC Network Filtered Output .....	27

**LIST OF TABLES**

TABLE 2.1.3.2 Look-Up Table Example .....	11
TABLE 2.2 Motor Parameters Associated w/ Measured Data Files.....	14
TABLE 3.2 Average RMS Errors for Input Redundancy .....	26
TABLE 3.4 Average RMS Errors for Different Estimation Methods .....	28

## 1.0 INTRODUCTION

The basis for artificial neural networks is a fascinating concept. The idea of modeling the processes of the human brain on a computer and training this model to learn and then make decisions sounds like science fiction to those unfamiliar with this area of artificial intelligence. Indeed, the scientific community is not convinced that an artificial neuron may develop into a viable component as applicable as the transistor. The future of this exciting concept is left to the philosopher of science for speculation. The intention of this paper is to promote an interest in the concept in the hope that more engineers will become enthusiastic about neural networks and pursue other initiatives for their application.

### 1.1 BACKGROUND: MORE ELECTRIC AIRCRAFT CONCEPT

The application of a neural network to a switched reluctance machine was conceived through the machine's crucial role in the More Electric Initiative (MEI). The vision of the More Electric Initiative is to replace conventional centralized hydraulic systems with fault tolerant electrical power in order to supply the aircraft loads. Through the implementation of this concept, life cycle costs will be reduced through improvements in component reliability. There will be a 30 to 50% reduction in Aircraft Ground Equipment (AGE), since hydraulic and pneumatic support carts can be eliminated. Major system level improvements are projected due to improvements of battle damage tolerance, maintainability, and supportability. Also, safety will be improved by the elimination of hydraulic maintenance procedures and the elimination of hydraulic fire hazards. Through exploratory development work in electrically powered flight controls, braking, and Environmental Control Systems (ECS), it has been shown that by using electrical power, these systems achieve an enhanced performance.

A Memo of Understanding has been signed to form a National Coalition for the More Electric Initiative. Members of this coalition include the Navy, Air Force, Army, and NASA. It has been projected that private industry will spend \$400 million on this program over the next eight years. The majority of these funds will be spent on Independent Research and Development (IR&D) programs. Existing programs involve the development of electric hydraulic actuators, electromechanical actuators, electric brakes, advanced Auxiliary Power Units (APUs), and Switched Reluctance Machines (SRMs).

In addition to industry funded MEI projects, there are a number of contracts being funded through the tri-services and NASA. For example, a number of contracts are ongoing or undergoing the process of being awarded that directly involve the development of switched reluctance machines. These contracts include a program entitled the Power Management And Distribution System for a More Electric Aircraft (MADMEL). The objective of this program is to develop an advanced electrical power generation and distribution system demonstrator. This contract has been awarded to Northrop Corporation; it is the contractor's intention to use switched reluctance machines for the

starter/generators in the demonstrator. Another pertinent contract is the Integral Starter/Generator Program. The objective of this program is to design a 250kW and a 375kW switched reluctance starter/generator. The 250kW generator will be a compact design that is externally mounted to the gearbox of the aircraft engine. The 375kW design is to be integrated around the shaft of the engine, resulting in the elimination of the gearbox. Also, another contract that is to be awarded calls for the design of a 125kW switched reluctance starter/generator to be built inside an APU. The functions of this machine include starting of the APU, emergency starting of the main engine, and supplying emergency power to the flight critical loads.

## 1.2 THE APPLICATION: SWITCHED RELUCTANCE MACHINE

The existence of a basic design for a switched reluctance machine can be traced back twenty years. Its commercialization had never been pursued successfully over the years due to the complexity of its hardware implementation. The power electronics required for its converter circuitry needed to be capable of switching high currents. Until recent developments in power electronics, these types of switches were not available [1]. With the development of the Insulated Gate Bipolar Transistor (IGBT) and the Metal Oxide Semiconductor Controlled Thyristor (MCT), converters have been successfully built. To date only a few military applications of the SRM have resulted in the development of prototype machines. *Examples of machines built include an integral starter/generator for an army tank and a switched reluctance motor for an aircraft actuator.* This section will discuss the basic principles of the switched reluctance machine design, its operation, and critical design issues.

### 1.2.1 MACHINE DESIGN & OPERATION

A cross section of a typical machine is shown in Figure 1.2.1.a. This machine has three phases with six stator poles and four rotor poles. The figure represents the simplicity of the electromagnetic machine design. This simplicity makes the machine robust in comparison to conventional machines such as wound rotor machines and permanent magnet machines. There are no windings required on the rotor, and only a few concentric coils are required on the stator poles. These stator windings can be connected in series or parallel depending upon the machine's converter design and its application. In Figure 1.2.1.a, only a single set of pole windings are shown for simplicity; the windings of opposite stator poles are connected in series to create a single phase. The other ends of the windings are connected between the two high power switches in the converter as shown in Figure 1.2.1.b. The angle of interest,  $q$ , is shown in Figure 1.2.1.a as the angle between phase A's stator pole and a rotor pole.

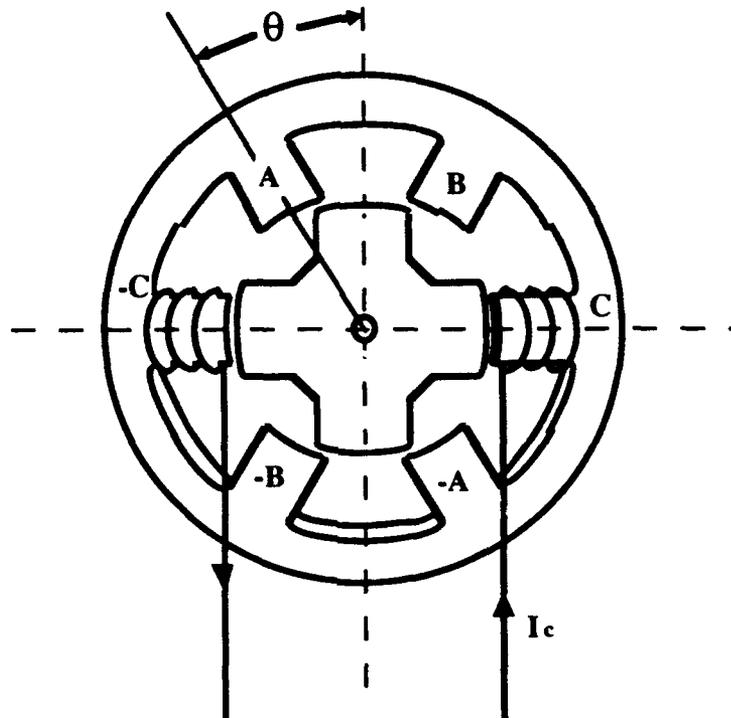


FIGURE 1.2.1.a Cross Sectional View of a Switched Reluctance Machine

The construction of both the rotor and stator cores consists of a laminated Vanadium Permendur material. The laminations are required in both the rotor and stator since the magnetic field will alternate in both parts of the machine.

A typical converter circuit is shown in Figure 1.2.1.b. It has two high power switching devices per phase leg [2]. The operation of the machine is controlled by the timing of the pulsing of these switches.

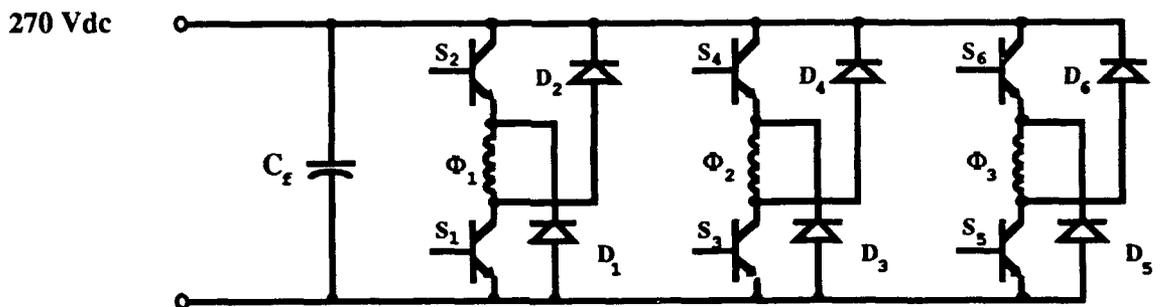


FIGURE 1.2.1.b Converter Circuitry

A significant advantage of this design is its built-in fault tolerance. Should one phase of the machine be disabled due to a short circuit or faulty switch, this phase can be de-excited and the remaining phases will be isolated and can continue to supply a reduced

amount of power. The power output under these conditions is reduced by the power contribution of the disabled phase [3]. The capability of continued operation under a fault condition is a unique feature of the switched reluctance machine.

The torque of the machine is developed through the magnetic attraction between the rotor poles and the stator poles. The timing of the magnetic excitation with respect to the relative rotor position controls the polarity of the torque pulses. Positive torque pulses are produced as the rotor poles are approaching the stator poles. The operation of the machine is shown in Figure 1.2.1.c. When the inductance has a positive slope, the machine will behave as a motor. The period during which the windings should be excited to achieve a maximum torque is shown in the figure as the ideal motor current pulse.

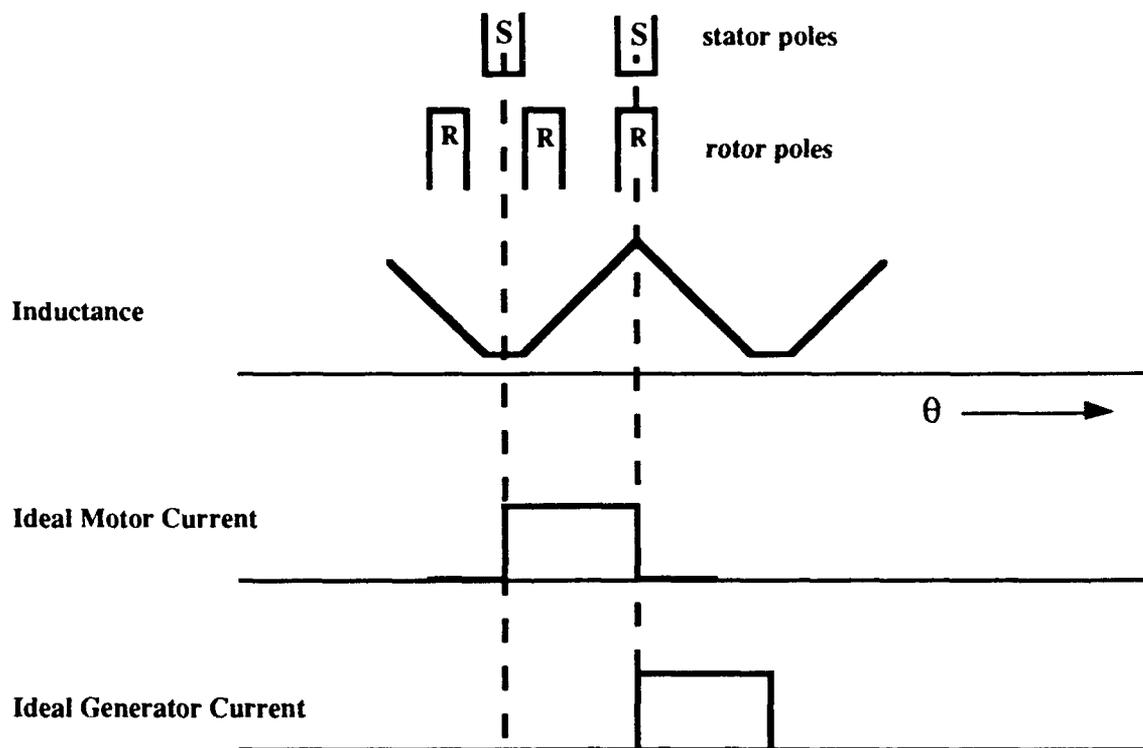


Figure 1.2.1.c Machine Operation

Alternately, as the rotor pole is leaving alignment with a stator pole, the torque pulses are negative. When the machine has a negative sloped inductance, it will operate as a generator. To achieve maximum electrical power extraction from the machine, the windings should be excited as the rotor pole is leaving alignment [4]. The angle period over which the machine is excited to operate as a generator is shown in Figure 1.2.1.c as the ideal generator current. In the description of the machine operation, the significance of the relative position of a rotor pole with the stator pole should be evident.

### 1.2.2 ADVANTAGES OF SWITCHED RELUCTANCE

The switched reluctance machine proves to be an attractive choice for many applications due to its high speed capability, its ability to operate in harsh environments, its good power density, and its fault tolerance.

Conventional aircraft generators are the wound rotor generator and the permanent magnet generator. In these conventional machines, the stator windings are closely wound and magnetically coupled. In the event of a short circuit in any one phase of the multiphase winding, an excessive amount of heat will be generated. If the machine is not immediately de-energized, the short may propagate to the remaining phases due to the tight interleaving of the stator coils [1]. In a permanent magnet machine, operation is further limited by the temperature limitations of the magnets. At high temperatures, the rare earth magnets used in these machines may lose magnetic stability or even become permanently demagnetized. On the other hand, the rotor of a switched reluctance machine is quite robust in that there are no windings nor permanent magnets. The rotor is a solid construction of laminated Vanadium Permendur. The stator windings consists of a few concentric coils; these coils are isolated from one phase to the other. Due to the simple construction of the SRM rotor and stator, it is capable of operating in much higher temperatures and speeds than conventional machines.

The permanent magnet machine has the highest power density of the three types of machines. The power density of a SRM is from 85% to 95% of the permanent magnet machine's power density depending upon machine size. This power density is still relatively high and the other benefits of the SRM will often override the superior power density of the permanent magnet machine when choosing a machine type for a particular application.

As described earlier, the SRM has an inherent fault tolerance due to its unique concentric stator windings. These advantages will make the SRM a superior choice for many future applications.

### 1.2.3 CRITICAL DESIGN ISSUES

Although efforts are being made to develop these machines, a number of design challenges must still be met. The high current switches in the converter circuitry employ a new technology with design and manufacturing issues yet to be resolved. The rotor iron losses for the SRM are higher than those associated with conventional machines. These losses can result in high temperatures which will limit the machine's speed capability. The development of advanced materials for the rotor structure continues to evolve. Also, the controller for these machines still present significant design challenges.

Specifically, the design issue that is being addressed in this research program involves the determination of the relative position of a rotor pole with a stator pole. This relative angle is a necessary input to the machine's controller. In traditional designs, the

relative angle would be measured with a rotor mounted encoder or resolver. This additional piece of hardware presents a reliability risk, especially when the machine is expected to operate in harsh environments such as during the application for an integral starter/generator. The position sensor also creates a single point of failure for the system.

An alternative design involves sending phase voltage and current values to the controller and inputting these values to a magnetic circuit model which would theoretically predict the relative angle. This design is being pursued at General Electric's Corporate Research & Development Division [7].

A schematic of this approach is given in Figure 1.2.3.a. For each stator pole of the machine, the flux linkage is estimated by integrating the difference between the measured phase voltage and the product of the measured phase current and the winding resistance. Also, the magnetomotive force (mmf) can be derived from the measured phase current. The derivation of the reluctances which are represented in the magnetic circuit model of Figure 1.2.3.a as rectangles, is more complex. These reluctances are analogous to resistances in an electrical circuit. They depend upon the machine geometry including such parameters as air gap distances, pole tip lengths, and stack thicknesses; they also depend upon the materials' permeabilities. These permeabilities are calculated by a saturable material model. Also, a finite element analysis is required for the calculation of the machine reluctances. The fluxes, mmfs, and reluctances are then incorporated into a magnetic circuit from which the magnetic model mesh equations are derived.

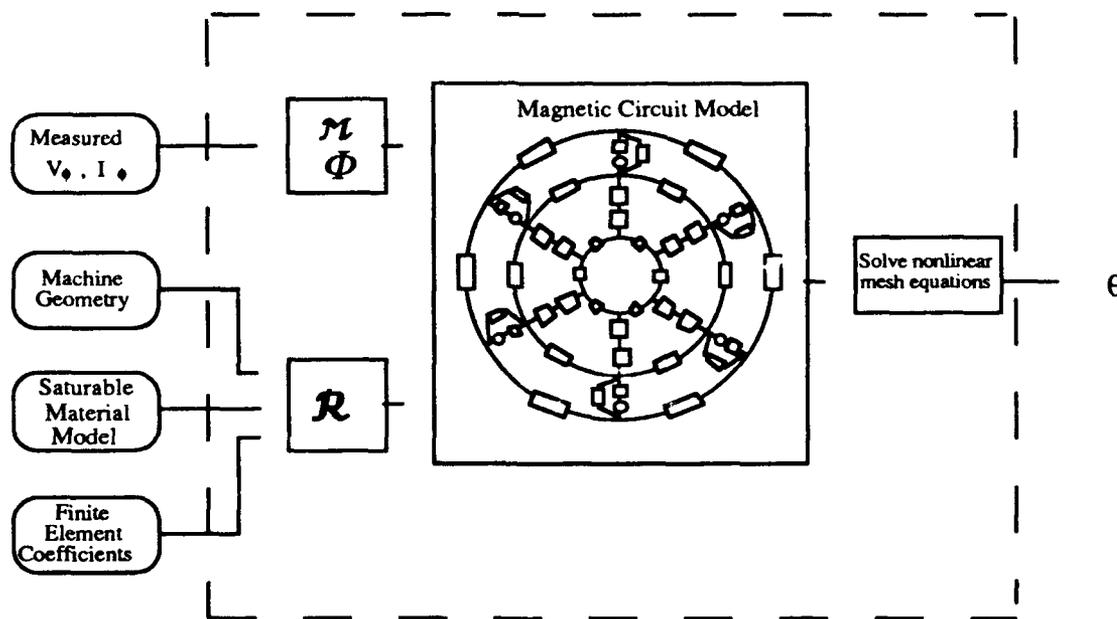


Figure 1.2.3.a Magnetic Model Approach

The airgap reluctance is the only variable which is a function of the rotor angle; therefore, this reluctance is isolated and the angle is solved for. This process must be undertaken for each set of measured voltages and currents that are sampled.

Certain problems with this approach exist. The computational power required to perform the calculations will require a Digital Signal Processor (DSP) which may be viewed as overkill for this particular problem. The extensive computation limits the performance capability of the machine. The delay associated with the computation will impact the control loops of the generator control unit and create additional design challenges. Also, the model's algorithm breaks down in the event of a fault condition. General Electric is currently addressing these design issues.

Another approach to the problem of determining the relative angle without using a position sensor is the basis of this research paper. This approach uses the same voltages and currents as the model approach uses. These parameters are used as inputs to a neural network (See Figure 1.2.3.b).

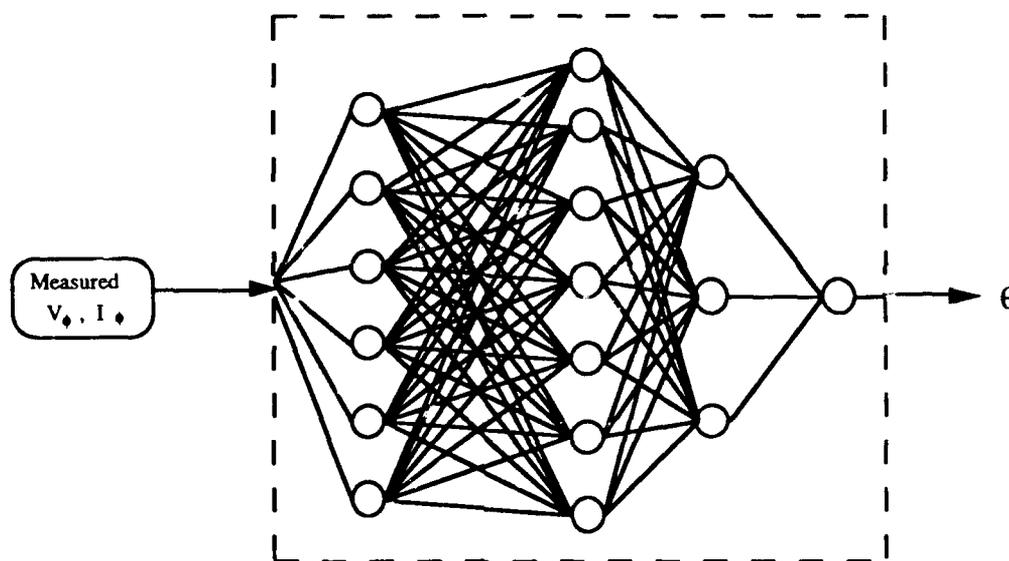


FIGURE 1.2.3.b Neural Network Approach

A neural network approach should require significantly less computational effort than the magnetic circuit model method. The operations involved in the mapping calculations are additions, multiplications, and simple look-ups as opposed to complex integrations, differentiations, and finite element analyses. Due to the simplicity of the mappings, the network should be able to provide a real-time response. As for operation of the network under fault conditions, neural networks are known to have an inherent system redundancy. Networks have been shown to correctly map even with faulty or missing input signals. For these reasons, the neural network approach should be the least

computational, most responsive, and fault tolerant approach for rotor position estimation.

## 2.0 METHOD

A number of different artificial neural network paradigms were investigated for the application of a rotor position estimator. It was concluded that the network would be a feedforward heteroassociative type. The candidate paradigms chosen were backpropagation, Radial Basis Functions (RBF), and Cerebellar Model Articulation Controller (CMAC). The results from these three types of networks are summarized in Section 3. A detailed description of the results from the backpropagation and radial basis function networks can be found in the Independent Study Report [6]. The best results were obtained with the CMAC network, so this network is described in detail in the following section.

### 2.1 THE PARADIGM: CMAC

#### 2.1.1 A HISTORICAL PERSPECTIVE

In 1972 a neurophysiological model was first described by James S. Albus in his Ph.D. Thesis at the University of Maryland. The theory behind the model implies that the structure which is responsible for reasoning and decision-making in the human brain is similar to the sensory/motor structure. Also Dr. Albus postulated that the reasoning process and sensory/motor process are not unique separate systems but belong to one interacting system. The functions of thinking, decision-making, sensing, and moving are all interdependent [7]. An simple example of this interdependence is when a child touches a hot stove, the sensory experience (the pain felt on his hand) is associated with the motor action (touching the stove) and is stored as a learning experience so that a decision can be formulated by his thought process (don't touch the stove).

Neurophysiological evidence shows that many biosensorimotor control structures in the brain are organized using neurons that possess locally-tuned, overlapping receptive fields. CMAC and Radial Basis Functions are two examples of artificial neural networks that also use these type of receptive fields.

The name of this model varies in the literature but its acronym, CMAC, remains the same. In earlier work, CMAC stood for the Cerebellar Model Arithmetic Computer and is described as a computing device which accepts an input and through a series of mappings will produce an output. At some point, as the model was applied to robotic control problems, its acronym inherited a new meaning; namely, the Cerebellar Model Articulation Controller.

Current work on CMAC is ongoing at the Intelligent Systems Group of the Robotics Laboratory at the University of New Hampshire and in the Human Information Processing Group of the Department of Psychology at Princeton University [8,9,10]. The

most active sponsors for this type of work are the Advanced Research Projects Agency (ARPA), the National Science Foundation, and the Office of Naval Research (ONR).

### 2.1.2 THE BENEFITS OF CMAC

CMAC is advertised as being capable of learning nonlinear functions extremely quickly. It offers an alternative to Multilayer Perceptron networks such as backpropagation. In backpropagation, all the weights are updated at each training presentation; this is called global training. In CMAC, only the weights selected by the training vector being presented are updated; this is called local training. Global training is more likely to distort the details of the borders of the classes, whereas local training allows one part of the input space to be trained without corrupting what has already been learned in other areas of the space. Global training also slows the rate of learning. Besides the obvious disadvantages of a slow learning rate such as long and expensive computer training times, a slow learning rate makes on-line learning unrealizable and also imposes the need to use small networks. For these reasons, globally trained networks can not be used for many complex problems. Global training also makes the accuracy of the output function sensitive to the presentation order of the training data.

CMAC utilizes the adaptive Widrow-Hoff Least Means Square (LMS) learning rule. The error surface is quadratic so that the search results in a unique minimum. One of the primary complaints with backpropagation learning, is the occurrence of many relative minimums. The search for the solution can become trapped in a relative minimum and the solutions never reached. The Widrow-Hoff learning rule updates the weights at each presentation of a new training vector by the following equation:

$$\delta w = (\beta/c)(y_{di} - w_o^T x_i),$$

where  $\delta w$  is the change in the weights,  $\beta$  is the training factor which can be adjusted between 0 and 1,  $c$  is the number of mappings,  $y_{di}$  is the desired output for the  $i$ th input vector,  $w_o$  is the original weight vector, and  $x_i$  is the  $i$ th input vector of the training set.

Since CMAC uses a linear output layer, it obeys superposition. By superposition, if a set of weights  $w_1$  produces the nonlinear function  $f_1(x)$  and the set of weights  $w_2$  produces the nonlinear function  $f_2(x)$ , then the set of weights  $w_1 + w_2$  will produce the function  $f_1(x) + f_2(x)$ . This property makes it possible to characterize the ability of a network to produce a class of functions of given dimension, weight size, and generalization. Superposition also permits the paralleling of many small CMACs to perform similar functions [8].

2.1.3 THE ALGORITHM

The method of implementing CMAC is detailed in the functional schematic, Figure 2.1.3.a. For simplicity, this figure shows only three inputs and one output, but these variables are adjustable. The inputs are  $x = \{x_1, x_2, x_3\}$  and the output is  $y$ .

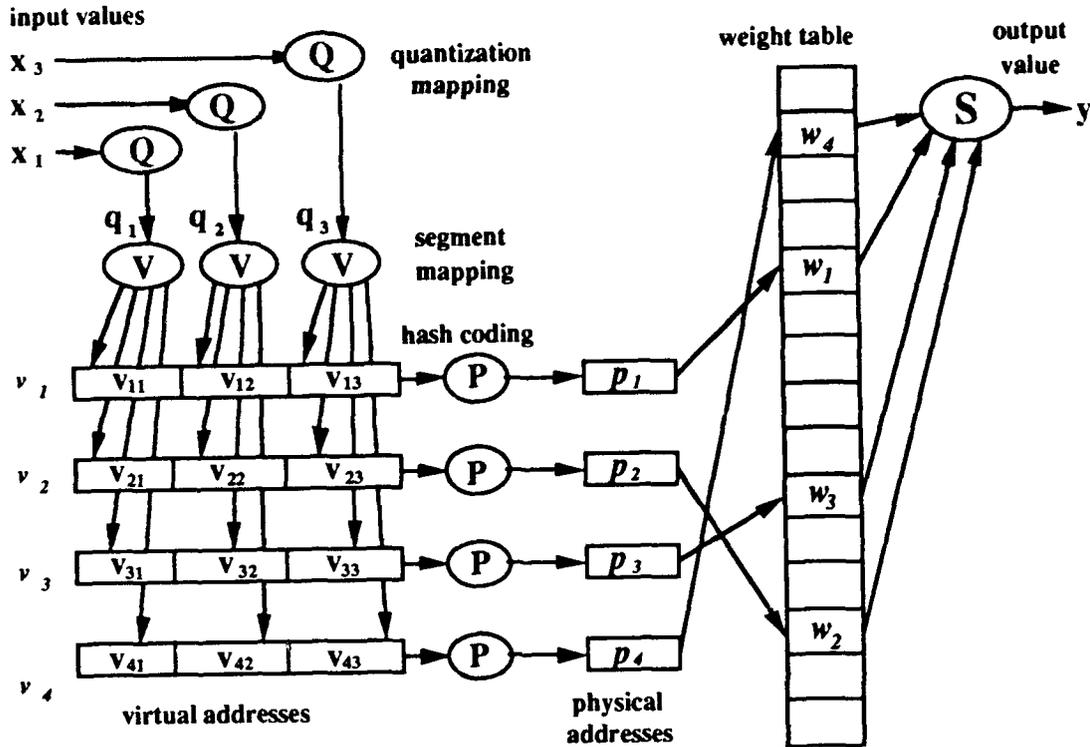


FIGURE 2.1.3 CMAC Mapping Scheme

The CMAC algorithm employs a series of intermediate mappings to convert the inputs to the desired output [11]. These mappings are referred to as input quantization, virtual address computation, hash coding, and output computation.

2.1.3.1 INPUT QUANTIZATION

The input quantization involves a normalization of the inputs based on the maximum and minimum values of the inputs. For this particular implementation, the inputs are always normalized between 0 and 1. A resolution,  $r$  is chosen such that each element of an input vector can be placed in a bin. For example, if the resolution is 1000, there are 1000 possible bins each of equal width into which the inputs can be placed.

2.1.3.2 VIRTUAL ADDRESS COMPUTATION

The next mapping occurs through a virtual address computation. A set of address

segments is computed for each element of the input vector. Each address segment is found in a look-up table. An example of such a table is shown in Table 2.1.3.2.

The number of columns in the table corresponds to the resolution chosen for the input quantization. The number of rows is equal to  $m$ , the number of mappings chosen for the network. The number of mappings is also the number of weights that will be updated upon each new presentation of an input vector.

The contents of the table are integer indices. The table is constructed by filling the rows from left to right and top to bottom with integers starting with an index of one. Moving through the table, the index is incremented by one after the number of positions in the table filled by that index reaches the overlap. The index is always incremented at the beginning of each row. This index is repeated for one column more than the above row's first index unless the above row's index was repeated for the full overlap in which case the index is only used in one column slot.

		← RESOLUTION →									
		1	2	3	4	5	6	7	8	9	10
NO. OF MAPPINGS	V <sub>1</sub>	1			2			3			4
	V <sub>2</sub>	5	6			7			8		
	V <sub>3</sub>	9		10			11			12	
	V <sub>4</sub>	13			14			15			16

$m = 4$   
 overlap = 3

TABLE 2.1.3.2 LOOK-UP TABLE EXAMPLE

The overlap is a variable that is chosen during the design of the network. The overlap determines the amount of generalization that occurs between the input variables. This variable is one of the unique features of CMAC. By having control over the amount of generalization, the boundaries of the classes or hyperspaces can be well controlled.

An address segment is assigned from the table in accordance with which bin the input element falls. Each input element will be associated with  $m$  address segments, a segment from each row of the table. After the address segments are computed for each element of an input vector, they are concatenated to create a single virtual address for the input vector [12]. Following a single iteration, there are  $m$  virtual addresses for each input

vector in the training set.

Table 2.1.3.2 can be used to show how a virtual address is computed and how the choice of the overlap will dictate the generalization of the network. Three quantized input vectors,  $q_a = \{1,5,9\}$ ,  $q_b = \{2,5,10\}$ ,  $q_c = \{8,1,3\}$  can be used to illustrate the address computation. For each quantized input vector, the corresponding sets of concatenated segments extracted from the look-up table are given below.

$q_a$ {123}	$q_b$ {124}	$q_c$ {311}
{578}	{678}	{856}
{91012}	{91012}	{11910}
{131415}	{131416}	{151313}

Since  $q_a$  and  $q_b$  were close in input space the overlap factor allows them to share a common virtual address segment, {91012}. Also note that vector  $q_c$  is not close to either  $q_a$  or  $q_b$  in input space and it will not share any common virtual address numbers.

### 2.1.3.3 HASH CODING PROCEDURE

The number of possible virtual addresses is  $mv^d$ , where  $v$  is the number of possible input values and  $d$  is the dimension of the input vector. For practical systems, this many addresses represents an unrealizable memory size. In fact, this many memory locations would not actually be used at one time. For these reasons, CMAC uses a hash coding to reduce the number of required memory locations.

For this particular implementation of CMAC, the method of hash coding uses a Random Number Generator (RNG); the RNG chosen utilizes the subtractive method. The virtual addresses are used as seeds to this RNG. The output of the RNG is a real number between 0 and 1. This number is then converted into an integer within the range of the number of weights chosen for the network. The number of weights is specified in kilobytes and it determines the size of the weight table. The weight table is a set of physical addresses in which the value of each weight is stored. For multiple outputs, the weight table is divided into blocks for each output, so that an independent set of addresses is available for each output variable.

The method of hash coding chosen greatly influences the performance of the CMAC network. In using a pseudo-random number generator, the creation of unwanted collisions is inevitable. The collisions occur when two unique seeds result in the generation of the same random number. When this happens, the same weight location is accessed when two unique locations should be accessed. This will result in a corruption of the weight values. Although these collisions are unavoidable, since no RNG is truly random, they can be minimized by using a RNG that maximizes the discreteness of the random values returned. During the implementation of this CMAC algorithm, different

RNGs were tested including the single linear congruential, the combination linear congruential, and the subtractive generators. The best results were obtained using the subtractive RNG.

#### 2.1.3.4 OUTPUT COMPUTATION

The output is computed as a linear summation of the weights associated with the input vector. Initially, the value of the weights are set equal to zero. Following the first presentation of data, the computed output will be zero, so that the weights affected by the input vector are adjusted by the desired output value.

Upon the second presentation of an input vector, the same process is performed in the quantization of inputs, the virtual address computation, and the hash coding to physical addresses. The output is then computed by summing the values stored in those weight locations which were identified by the hash coding. If the second input vector is far in input space from the first input vector and no collisions occur during the hash coding, then this summation will again equal zero and those weight values will be adjusted by the desired output. On the other hand, if the two input vectors are close in input space or if collisions occur from the hash coding procedure, then some weight locations will be shared. This sharing results in a non-zero computed output and the weight locations will be adjusted by the difference of this computed output and the desired output.

#### 2.1.4 HARDWARE IMPLEMENTATION

CMAC can be practically realized in hardware by the use of logic cell arrays. VLSI versions are feasible. An example of a product that is currently available is manufactured by Shenandoah Electronic Products of Newington, NH [13]. Their CMAC-AT is a memory board for an IBM PC-AT compatible computer. The board is made-up of CMOS field programmable gate arrays. The board can be configured for up to 8 independent networks. It holds one million 8 bit adjustable weights or 512K 16 bit adjustable weights. The size and number of inputs and outputs and the overlap are all adjustable parameters. Typical response times for a network with 32 integer inputs and 8 integer outputs are on the order of 200 to 500 microseconds.

#### 2.1.5 SUMMARY

The significant properties of CMAC can be summarized as follows. CMAC accepts real inputs and provides real outputs. Even though the inputs are quantized, the resolution is adjustable so that any degree of accuracy is possible within the memory limitations of the computer. CMAC maps with local generalization. In other words, input vectors that are close in input space will result in outputs that are close in output space. CMAC has the property that large networks can be trained in practical time. This is due to the fast learning

rate or its quick convergence and due to the fact that there are a small number of calculations per output even if there are a large number of weights. CMAC uses the LMS learning rule of Widrow and Hoff. This algorithm uses a gradient search which has a unique minimum so that the problems associated with backpropagation and relative minimums are non-evident. CMAC obeys superposition in the output space, and it has a practical hardware implementation.

**2.2 DATA PREPROCESSING**

The data for training and testing the network was obtained from a 120 hp, 6/4 pole switched reluctance motor. Motor parameters measured included three phase voltages, three phase currents, the rotational speed and the relative rotor angle. The phase voltages and currents were measured with existing sensors in the machine's converter circuitry. The speed and angle were measured using the conventional resolver of the machine. The data was recorded using a data acquisition system with a 25 KHz sampling rate and a 16 bit resolution. Since the data was measured with a data acquisition system, the discrete points were recorded as integers over the range +/- 32768. For interpretation purposes, the data was converted to real numbers in units of amps for current, volts for voltage, rpm for speed, and mechanical degrees for angle. (These are the units used in Figures 2.2.a - 2.2.c.) The voltages and currents are used as the six inputs to the neural network and the rotor angle is the desired output. The rotational speed was not used as an input, since it is derivable from the rotor angle for a constant sampling rate.

Six files of data were recorded; each containing information under different motoring operating conditions. The differences in the motor operations for each file are listed in Table 2.2. The operating voltage of the machine was 100 Volts Direct Current. Data was taken under two operating speeds, 7000 rpm and 15,000 rpm. The loads applied to the motor resulted in current drawn ranging from 75 Amps to 300 Amps.

File Name	Rotational Speed (rpm)	Voltage Level (Volts)	Peak Current Drawn (Amps)
M7K100V.075	7000	100	75
M7K100V.150	7000	100	150
M7K100V.225	7000	100	225
M7K100V.200	7000	100	300
M15K100V.200	15000	100	200
M15K100V.300	15000	100	300

**TABLE 2.2 Motor Parameters Associated With Measured Data Files**

Preprocessing of the raw data was necessary due to an excessive amount of measurement noise. The preprocessing procedure became quite time consuming. Alternative strategies for processing the data in order to optimally train the network were examined. The data processing procedures performed on the raw data included reordering, stripping, filtering, and normalizing. The order of the implementation of these processes did not impact the performance of the network. However, each process proved necessary in achieving a sufficient data set for training.

Each data file contained voltage and current waveforms measured over a large number of rotor revolutions under varying speed and load conditions. An example of the each data column plotted versus its record number is shown in Figure 2.2.a. There were 4095 records in each file, but only the first 250 records are plotted in Figure 2.2.a.

The first preprocessing procedure involved a reordering of the data. The data was provided in the order that it was measured. In order to get a better visual grasp of the data, the records of the file were rearranged in order of increasing rotor angle. Figure 2.2.b shows each data column plotted versus the record number in its new order.

The next procedure termed stripping, deleted all unnecessary or unusable data from each file. First, the column containing the speed data was removed since this information would not be available for the neural network to use as an input. Next, the first period of the waveforms for each column was removed because the noise from this data was much greater than the other periods. Of the remaining three periods of data, if a single data point was thought to be capable of resulting in an unrealistic distortion of the waveform after it had been filtered, it was also removed. An example of the stripped data plotted as columns versus record number is shown in Figure 2.2.c.

The next procedure involves a conversion of the angle data from mechanical degrees to electrical degrees. The data files at this point contain three electrical periods, but the angle column ranges from 90 to 360 mechanical degrees. The angle values are thus converted into three sections of data from 0 to 360 electrical degrees.

Next, all data is normalized since the CMAC module requires normalized inputs. Each maximum and minimum value of the voltage, current, and angle waveforms is found. Then, these waveforms are normalized between 0.0001 and 0.9999 by the following formula:

$$y_{new} = scale \times y_{old} + offset$$

where  $y_{old}$  is the original data point,  $y_{new}$  is the normalized data point, and

$$scale = (0.9999 - 0.0001) / (max - min)$$

$$offset = (0.0001 \times max - 0.9999 \times min) / (max - min).$$

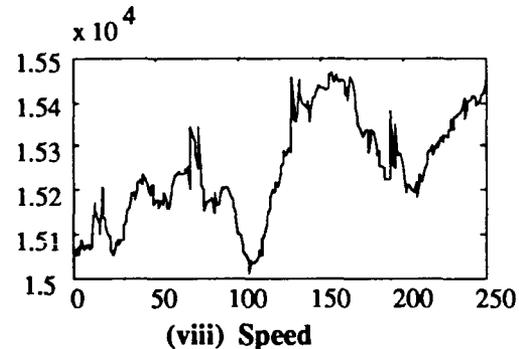
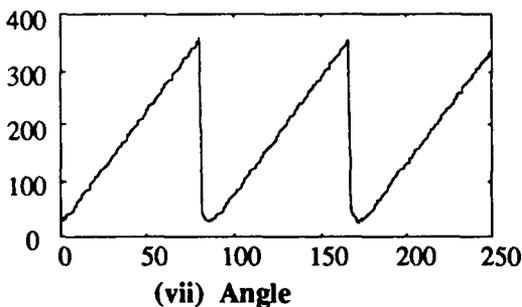
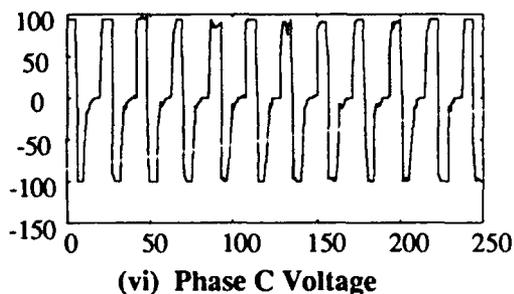
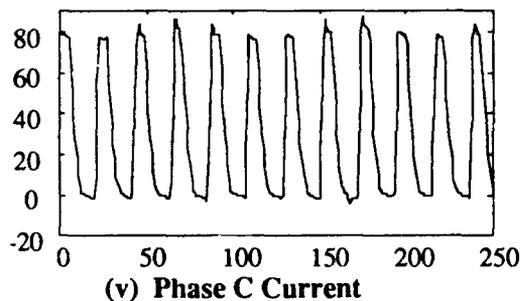
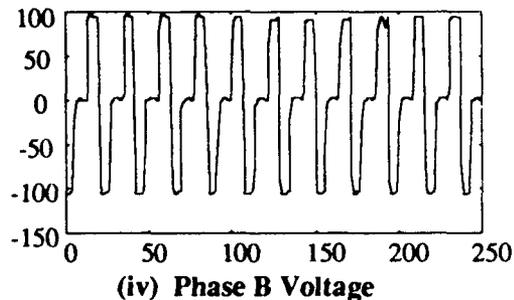
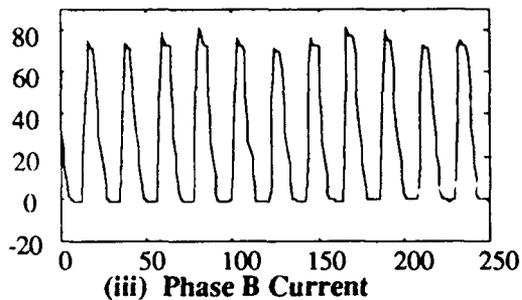
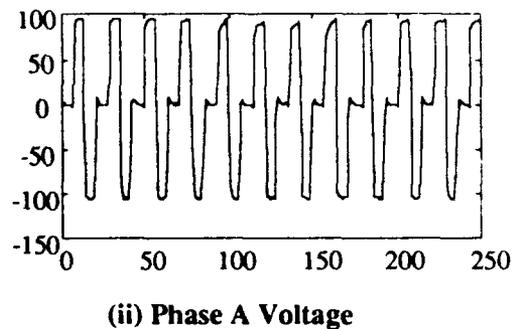
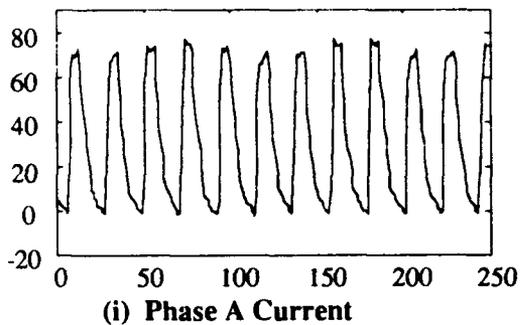


FIGURE 2.2.a Data Columns Versus Record Number

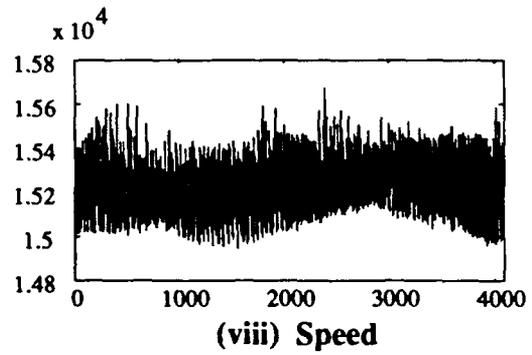
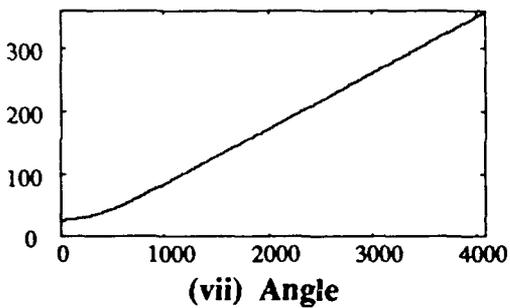
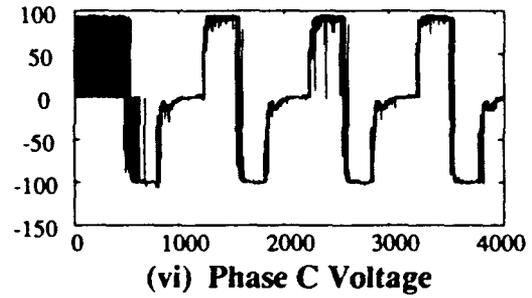
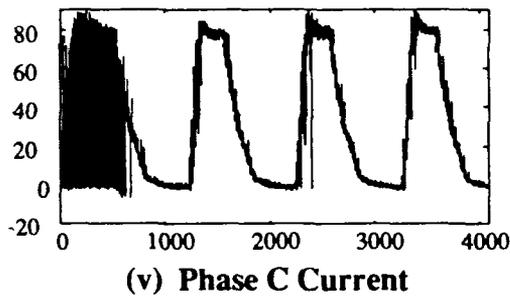
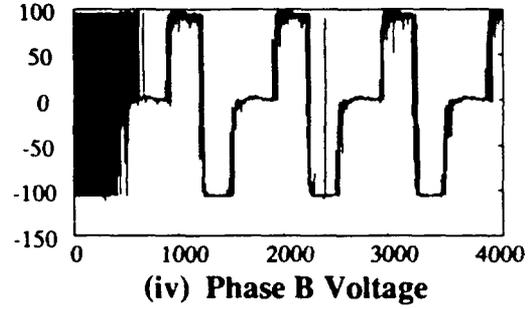
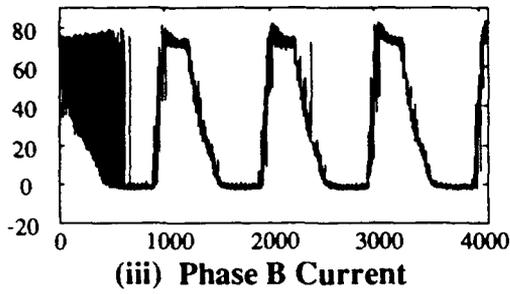
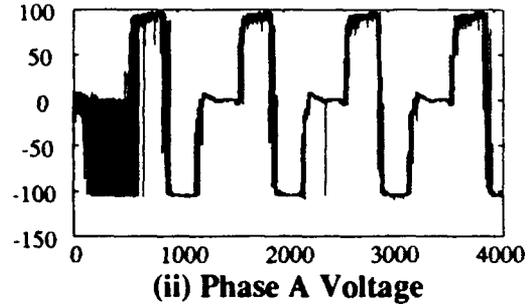
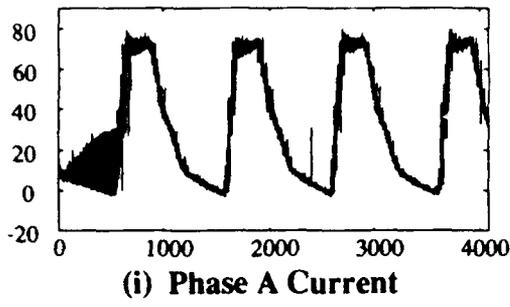


Figure 2.2.b Reordered Data Columns versus Record Number

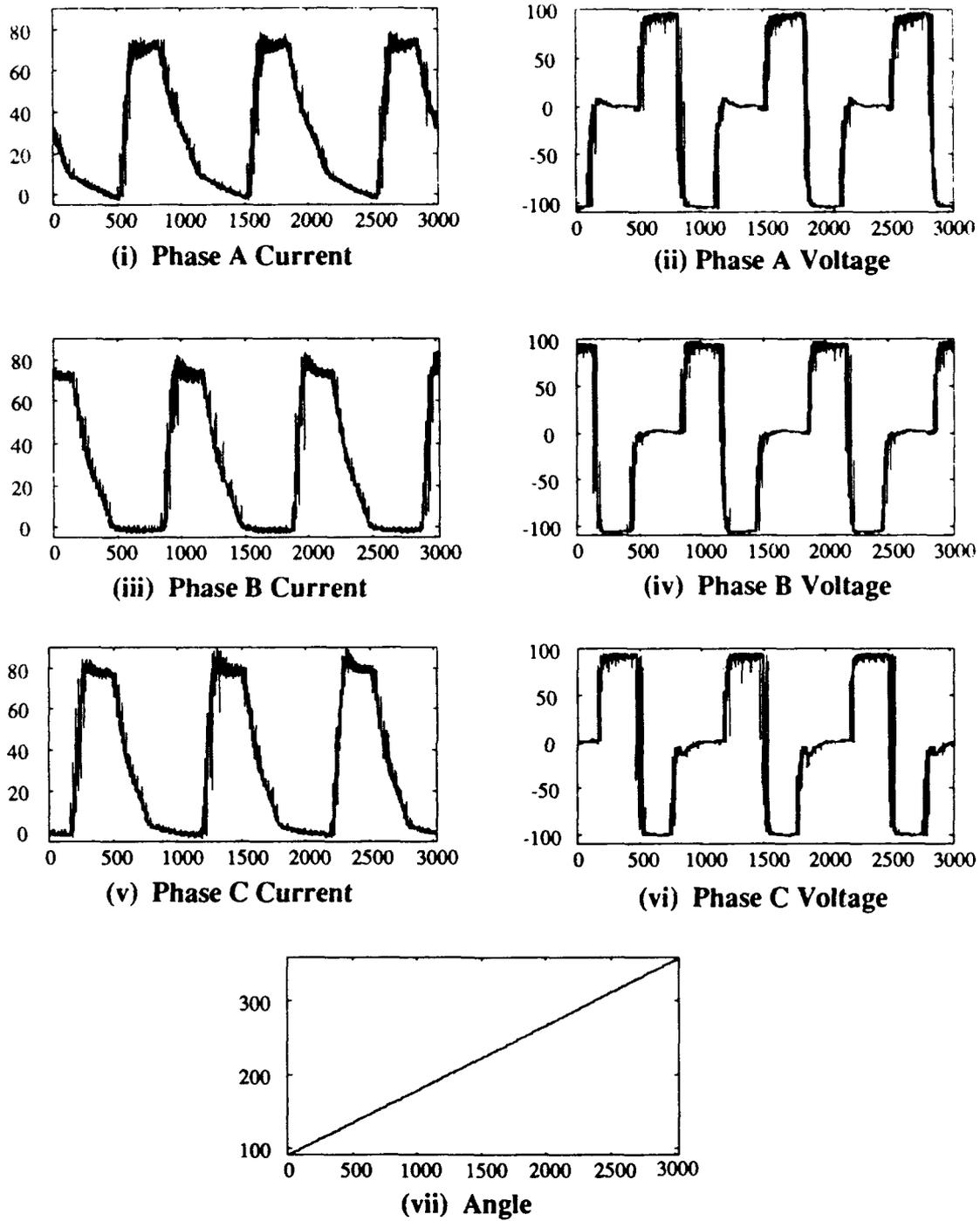


FIGURE 2.2.c Stripped Data Columns versus Record Number

The next procedure performed is a filtering process. A number of digital filters were tested. It was discovered that a filter resulting in the smoothest waveforms did not result in an optimal training set for the neural network. The more distinguishable the waveforms, the better the network's performance. The final choice for a filter was a 10-point finite impulse response low pass filter with a stopband cutoff frequency of 0.1 Hertz. An example of the normalized and filtered waveforms is plotted in Figure 2.2.d.

The final procedure is a separation of the data into training and testing data. In order to assess the network's ability to generalize, different data must be used for training and testing. A single training vector consists of the desired output - the electrical angle and the six inputs - three currents and three voltages. The training file consists of every other vector in the filtered data files. The remaining points make up the testing file. All of the processed data used for training is plotted as input verses output. These graphs are included in Appendix B.

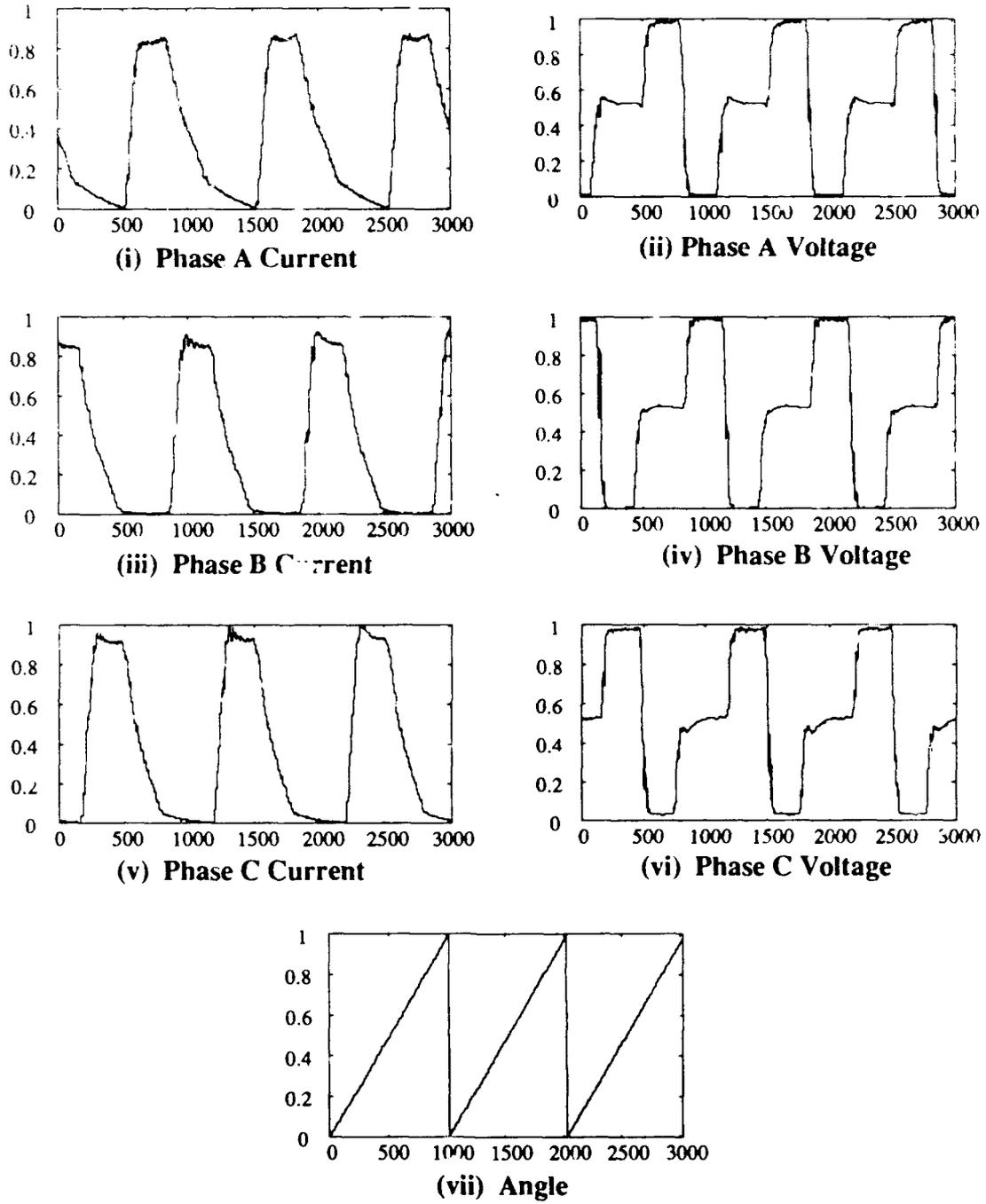


FIGURE 2.2.d Normalized and Filtered Data Columns versus Record Number

### 2.3 THE IMPORTANCE OF CMAC PARAMETERS

In the implementation of CMAC, there are a number of design parameters that can be adjusted for different applications of a CMAC network. These parameters are the following: weight table size, number of training vectors, resolution, overlap, number of mappings, training factor, minimum average root mean square (rms) error, minimum delta error, number of iterations, weight adjustment tolerance.

The weight table size is the number of kilobytes that will be allocated by the program to store weight values. The actual number of locations that will be available for storage can be calculated as follows:

$$\text{number of locations} = (\text{weights\_table\_size} \times 1024) / \text{sizeof(float)}.$$

The size of the floating point number is dependent upon the machine on which the program is running and the compiler. During this investigation, the program was run on both a Macintosh SE/30 and a Sun 630 MP Workstation. In both cases, the size of the floating point is 4 bytes. So, for example, if the weight table size chosen is 1 megabyte, then the number of locations available for weight storage would be 262,144.

The number of vectors used for training,  $n$ , should be optimized. In general, the more vectors used for training, the better the network will generalize. However, if the vectors are closer in input space than the required accuracy of the network, then not every vector needs to be used for training. If the unnecessary vectors are removed from the training set, then the training time can be reduced.

The resolution is the number of bins that will be available for the quantization of the inputs. The greater the resolution, the greater the accuracy of the network mappings.

The overlap is the number of bins that will be grouped with the same index in the look-up table during the virtual address computation. The overlap can be thought of as the width of an individual neuron or the size of the neuron's neighborhood. When adjusting the resolution or the overlap, the other parameter should be taken into consideration because the relationship between these two parameters determines the amount of generalization that the network is capable of performing. It should be noted that the size of every neuron is not necessarily equal to the overlap. (See discussion of Table 2.1.3.2).

The number of mappings chosen is the number of weights that will be used in the calculation of the output for a given input vector. By increasing the number of mappings, a greater percentage of the weight table will be used and a greater number of collisions will occur.

*The training factor is used to accelerate or decelerate the learning process. If the process is too fast, the errors may overshoot and continue to increase out of control. The highest tolerable training factor should be used in order to minimize training time.*

The program can be exited in three possible ways: by reaching the minimum

average rms error, the minimum delta error, or the maximum number of iterations. Whichever of these parameters is reached first will result in a termination of the training. The average rms error is calculated as follows:

$$\text{err}_{rms} = \frac{\sqrt{\sum_{i=1}^n (\text{out}_{ic} - \text{out}_{id})^2}}{n}$$

where  $\text{out}_{ic}$  is the sum of the weights associated with the  $i$ th input vector, and  $\text{out}_{id}$  is the desired output associated with the  $i$ th input vector. A running sum is kept of the squared difference between the computed and desired outputs over the entire training set. The square root of this sum divided by the number of training vectors is the average rms error.

The delta error is simply the difference between the average rms error of the previous iteration and the average rms error of the current iteration. When the delta error becomes very small, the weight adjustments are near negligible and the performance of the network will not be improved with additional iterations.

The delta error and the rms error are good indications of the training performance. If these numbers are steadily decreasing, the network is training successfully. The errors are also indications of the network's mapping performance. If the errors are small, the network will be capable of recreating the results of the vectors used in training. The ability of a network to reproduce mappings with which it is taught is termed the network's recall capability.

However, the errors are not good indications of how well the network will generalize. Generalization must be measured by testing the network with unique vectors that were not used for training. The convergence of a CMAC network depends upon the smoothness of the function being trained within the neighborhood over which generalization occurs. If the function varies greatly or is discontinuous within a generalization neighborhood, the rms error may not converge to an acceptable value.

The maximum number of iterations is self-explanatory. This parameter is useful for running batch jobs overnight. For example, if a single iteration is known to take one hour, by setting the number of iterations to twelve, the results will be ready twelve hours later.

The weight adjustment tolerance is the minimum difference between the computed output and the desired output that will warrant a weight adjustment. By including this tolerance in the training process, unnecessary adjustments can be avoided.

A characteristic of the network which is determined at the termination of the program is the percentage of memory used for the weight table. If this percentage is high, there are probably too many unwanted collisions, and the size of the weight table should be increased or the number of mappings should be decreased.

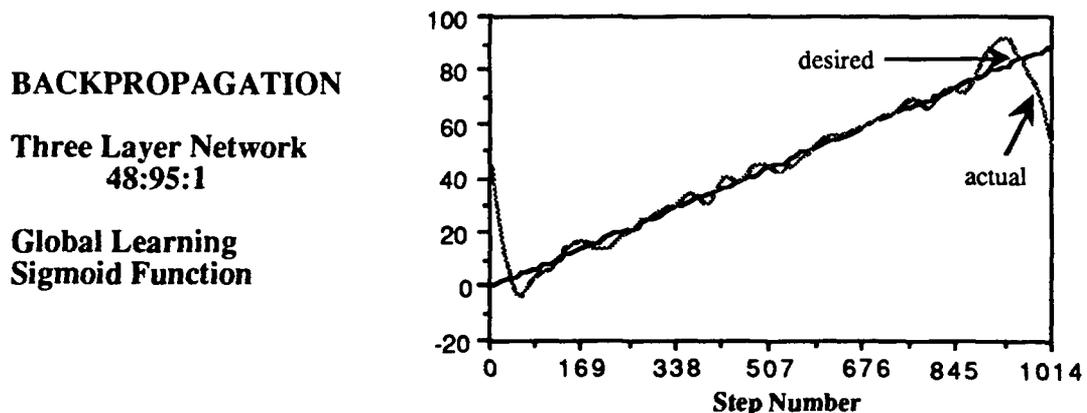
### 3.0 RESULTS

In order to initially test the feasibility of possible network paradigms, a single set of data was selected to train the different networks. At first, in order to select the paradigms with the most promise, only the networks' recall capability was examined. If the recall capability proved satisfactory, then its generalization capability was examined.

#### 3.1 COMPARISON WITH OTHER PARADIGMS

Three types of networks were identified as possible paradigms: backpropagation, radial basis functions, and CMAC. A great deal of effort was extended on the backpropagation method. Obtaining meaningful results proved to be quite time consuming due to the long training times associated with backpropagation. Networks could train for a period of three to four days before useful results could be analyzed. The backpropagation method's recall capability was extremely poor without the use of past input information. With 3 currents and 3 voltages, there are 6 instantaneous inputs available. However, the network required knowledge of the seven previous measurements ( $t, t-1, t-2, \dots, t-7$ ) in order to predict the desired output. For this reason, there are 48 inputs to the network. In general for the backpropagation paradigm, the more complex the mapping, the more neurons or elements are required for the middle layers of the network.

With the algorithm used in this research, the maximum number of elements available was ninety-five. The middle layer and output layer utilize a sigmoid function. Figure 3.1.a shows this backpropagation network's recall capability. The measured output of the network is plotted in comparison with the ideal output. The network's performance worsens at the beginning and the end of the electrical cycle. This poor performance is attributed to the sharp discontinuity here in output space.



**FIGURE 3.1.a Backpropagation Recall Capability**

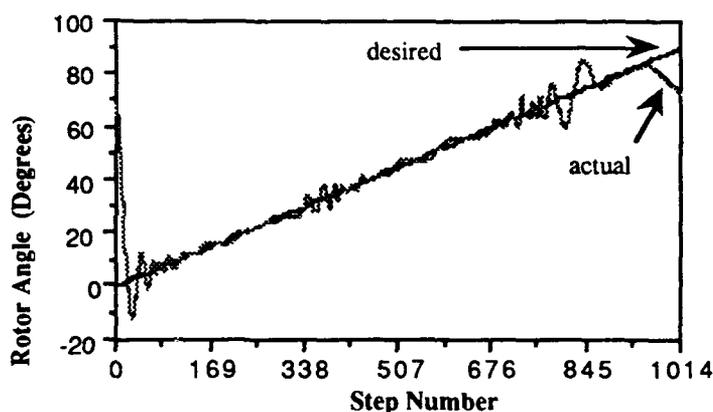
The radial basis function network is quite different from the backpropagation network, in that the output layer is a simple summation. This network structure allows training to be performed by a simple linear regression technique. This method of training is not nearly as time consuming the backpropagation method, so results can be obtained relatively quickly.

The RBF network is trained locally, rather than globally. In other words, each neuron in the middle layer of the network covers a portion of the input space. A random technique is used to determine the center of each neuron and its width. This technique works quite well if the network's input data is uniformly distributed over the input space. However, since the data for this particular application is not uniform, this technique proved to be less than satisfactory. Alternative techniques for determining the neurons centers are plausible, but would require considerable effort to implement. Prior to pursuing these avenues, alternative paradigms were investigated. The recall performance for the RBF network is shown in Figure 3.1.b by plotting the measured network output with the desired output.

**RADIAL BASIS  
FUNCTION**

**Three Layer Network  
6:100:1**

**Localized Learning  
Gaussian Function**



**FIGURE 3.1.b Radial Basis Function Recall Capability**

The next paradigm investigated was the CMAC network. Like the RBF network, training time was relatively short as compared to backpropagation. Also, as shown in Figure 3.1.c, CMAC's recall capability far exceeded the capabilities of the other paradigms; the difference between the desired output and the network's measured output are indistinguishable. Following the discovery of this network's superior capability, all research efforts were redirected to focus on this particular paradigm.

CMAC  
 Three Layer Network  
 6:2048:1  
 Localized Learning  
 Mapping Function

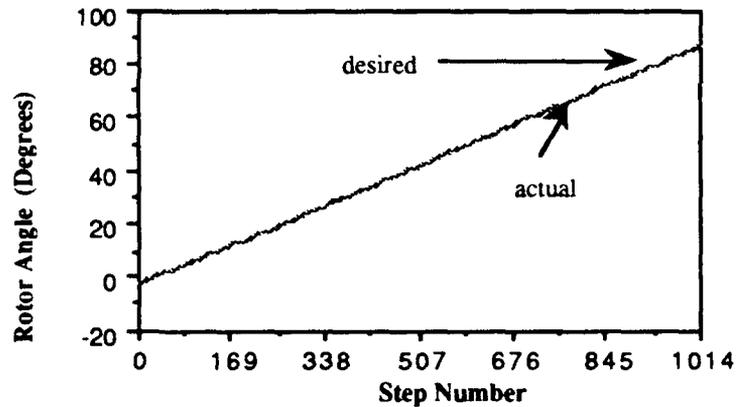


FIGURE 3.1.c CMAC Recall Capability

### 3.2 INPUT REDUNDANCY

As described in Section 1.2.3, the fault tolerance of the switched reluctance system is a critical design issue. It was noted that in the magnetic circuit model approach for rotor position estimation, the algorithm failed when the machine was under fault conditions. A major advantage of switched reluctance is the isolated winding structure which allows continued operation under fault condition, so it would be extremely detrimental to the system design if the estimation technique eliminates this advantage. It was anticipated that a neural network approach may be capable of providing a correct rotor position estimation given a faulty or missing sensor. At this stage in the network design, an investigation into this claim was initiated.

The maximum number of available inputs to the network for this application is six. A number of different combinations of two, three, and four inputs were tested. It was determined that only three inputs (half of the available inputs) were required for a successful mapping. The only restriction is that two of the inputs must be current waveforms; the third waveform may be either the third current waveform or any of the voltage waveforms.

An example of the accuracy achievable with only three inputs is shown in Table 3.2. The two test cases compared in the Table were trained with six inputs and with three inputs. The data used for training was from a single electrical cycle of a single motor parameter case. Each case was trained on the same CMAC network for ten iterations. In general, no significant difference between the performance of the network trained with three inputs versus six inputs was observed.

TABLE 3.2 Average RMS Error for Input Redundancy

Number of Inputs	Recall RMS Error (Mech. Degrees)	Generalization RMS Error (Mech. Degrees)
3	0.001032	0.002386
6	0.001257	0.002365

### 3.3 OPTIMIZED CMAC NETWORK

Following the observation of CMAC's excellent recall performance, its generalization capability was then assessed. As stated earlier, a unique feature of the CMAC allows control of the amount of generalization via adjustment of network parameters. There is a tradeoff, however, between the amount of generalization and network resolution.

In working with the other network paradigms (backpropagation and RBF), only data from one electrical cycle from one case of motor parameters was used for the training set. Adding the five sets of waveforms measured under different motor parameters to the training set would make each network's mapping increasingly more complex. A new waveform set (using three electrical cycles in each set) was added to the training set for the CMAC network one at a time. The network's parameters were adjusted accordingly, until all six sets of waveforms had been included.

Since only three of the four electrical cycles were uncorrupted by the data measurement process, only data from these three cycles were used for training and testing. So, Figure 3.3.a shows the output of the CMAC network for three cycles of the six sets of data (eighteen periods). Note, the Figure shows the network's generalization capability since the data used for testing was different than the data used for training.

The calculated average rms recall error was only 2 one thousandths in mechanical degrees and the generalized rms error was a tenth of a degree. However, the maximum errors are large at the discontinuities in output space as was evident when measuring recall capability in the backpropagation and radial basis function networks. Using a Finite Impulse Response (FIR) filter, the average errors can be slightly reduced. Figure 3.3.b shows the filtered output.

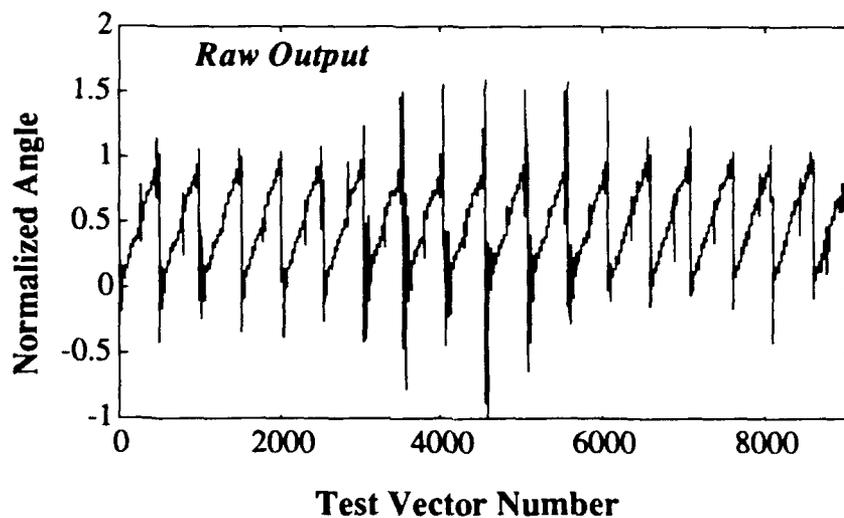


FIGURE 3.3.a CMAC Network Output

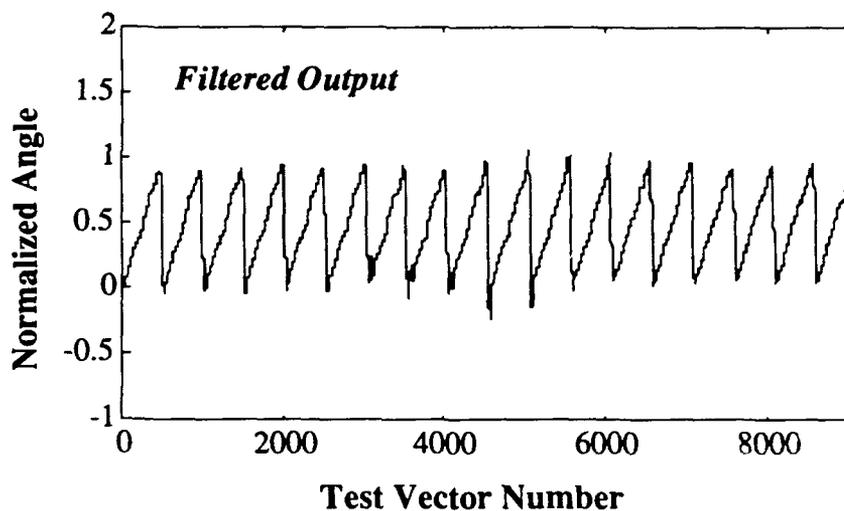


FIGURE 3.3.b CMAC Network Filtered Output

The FIR filter used was a Hamming-window lowpass linear phase filter with a cutoff frequency of  $1e^{-6}$  and an order of twenty-nine. The filter is applied in a non-causal manner that produces no phase distortion and minimizes startup transients. Filter design and implementation were performed using the Signal Processing Toolbox from the MATLAB software package.

### 3.4 DISCUSSION

The rms error for the relative position of a rotor pole with a stator pole is usually provided in mechanical degrees which generically is more meaningful. Electrical degrees are of particular interest for specific machine designs. In the machine used for this research, since there are four rotor poles, the electrical period is essentially ninety mechanical degrees and the rms error is four times greater in electrical degrees.

The errors are shown in Table 3.4 for the conventional rotor-mounted position sensor, the results reported by General Electric with their magnetic model method and the results achieved by this neural network method. It should be noted that the General Electric numbers were calculated with only three points per electrical period; the neural network numbers were calculated with over one thousand points per period. The true representation over the entire electrical period of this error over is not clear with only three points sampled per cycle. The neural network approach appears to offer a more accurate estimation of the relative angle over the other two approaches.

**TABLE 3.4 Average RMS Errors for Different Estimation Methods**

<b><u>METHOD</u></b>	<b><u>RMS ERROR</u></b> ( Mechanical Degrees )	<b><u>RMS ERROR</u></b> ( Electrical Degrees )
<b>Rotor-mounted position sensor</b>	<b>0.3</b>	<b>1.2</b>
<b>Magnetic Model</b>	<b>—</b>	<b>4.8</b>
<b>Neural Network (unfiltered)</b>	<b>0.116</b>	<b>0.465</b>
<b>Neural Network (filtered)</b>	<b>0.097</b>	<b>0.388</b>

It should be noted that the difference in the average RMS error between the filtered and unfiltered neural network outputs is not substantial. A systematic study of the required accuracy for the specific switched reluctance machine design should be performed. The additional output filter should only be incorporated into the network's design if such a study warrants.

### 3.4 CONCLUSIONS

An investigation of the application of neural networks for switched reluctance rotor position estimation resulted in the identification of an optimal network design based upon the Cerebellar Model Articulation Controller. The performance of this CMAC design far exceeded the performance of the other feedforward networks examined.

The results of this design was compared to the results of other approaches to rotor position estimation, namely a position measuring device and the magnetic circuit model method. Based upon the results comparison of the average rms errors between the different approaches to rotor position estimation, the neural network approach appears to provide the most accurate estimation. The CMAC network was modeled on a personal computer and a Sun workstation. The time required for a single mapping was a fraction of a second. If the CMAC network is implemented in hardware, this mapping time could be substantially reduced [14]. The computational effort required for the network is minimal, and initial investigation into the network's redundancy capabilities is promising.

Future efforts will be directed towards obtaining additional motor parameter cases and examining the effects of extreme motor conditions upon the voltage and current waveforms and the network performance. Future efforts will also be required to determine the feasibility of the neural network approach to estimate rotor position during motor starting.

Input redundancy will be further analyzed. In its initial investigation, the networks were trained to expect only three of the six inputs and then map the correct output. In future tests, the network will be trained to expect six inputs, but three of those inputs will be intentionally corrupted so as to more realistically simulate a fault condition. If measured machine data from a fault condition is obtainable, it will be used in this capacity. If continued success is noted with this research, the next step will be to investigate a hardware implementation of the CMAC design.

The application of neural networks to this particular problem demonstrates the viability of neural networks beyond applications of pattern recognition and signal processing. It is hoped that these efforts will continue and that the benefit of applying neural networks to engineering problems will flourish.

## REFERENCES

- [1] Richter, E. 1989. Application Considerations for integral gas turbine starter/generator revisited. Society of Automotive Engineers Technical Paper Series 892252 (Aerospace Technology Conference and Exposition, Sept. 25-28, 1989).
- [2] Finch, J.W., Harris, M.R., Mallick, J.A., and Miller, T.J. 1986. A review of the integral-horsepower switched reluctance drive. IEEE Transactions on Industry Applications IA-22: 716-21.
- [3] Boghbelli, H., et al. 1989. Comparison of theoretical and experimental performance of a 10 hp switched reluctance motor. IEEE Conference Rec. (IAS Annual Meeting, Oct. 1989) 89-98.
- [4] Bicknell, W.H., Bose, B.K., Miller, T.J., and Szczesny, P.M. 1986. Microcomputer control of switched reluctance motor. IEEE Transactions on Industry Applications IA-22: 708-15.
- [5] Lyons, J.P. and Preston, M.A. 1991. A switched reluctance motor model with mutual coupling and multi-phase excitation. IEEE Transactions on Magnetics 27 (November 1991).
- [6] Shannon, J.M. 1992. An investigation of switched reluctance rotor position estimation using neural networks. Report No. NAWCADWAR-92022-60.
- [7] Albus, J.S. 1979. Mechanisms of Planning and Problem Solving in the Brain. Mathematical Biosciences 45: 247-293.
- [8] Glanz, F.H., Kraft, L.G., and Miller, W.T. 1990. CMAC: An associative neural network alternative to backpropagation. Proceedings of the IEEE 78 (Oct. 1990).
- [9] Gelfand, J.J., Handelman, D.A., and Lane, S.H. 1991. Higher-order CMAC neural networks - Theory and practice. IEEE American Control Conference (June 1991).
- [10] Gelfand, J.J., Handelman, D.A., and Lane, S.H. 1992. Theory and development of higher-order CMAC neural networks. IEEE Control Systems (April 1992).
- [11] Gelfand, J.J., Handelman, D.A., and Lane, S.H. 1990. Integrating neural networks and knowledge-based systems for intelligent robotic control. IEEE Control Systems (April 1990).
- [12] Holl, L. 1992. Development and implementation of a cerebellar model articulation controller software module. Memorandum No. AVCSTD-92001-601.
- [13] Burgin, G. Using Cerebellar Arithmetic Computers. AI Expert (June 1992).
- [14] Campagna, D.P. and Kraft, L.G. 1990. A comparison between CMAC neural network control and two traditional adaptive control systems. IEEE Control Systems (April 1990).

**APPENDIX A**

**GLOSSARY**

**airgap:** A separating space between two parts of magnetic material which serves as a path for magnetic flux.

**alignment:** The point at which a rotor pole is lined-up with a stator pole and the magnetic circuit reaches saturation at its rated phase current and the maximum inductance is reached.

**backpropagation:** A neural network model consisting of a number of layers of neurons. The hidden layers utilize a sigmoid transfer function to model the nonlinearities of the mappings. The model uses a learning technique which can result in an undesired network convergence to a local minimum.

**Cerebellar Model Articulation Controller (CMAC):** a mathematical neurophysiological model of the cerebellar cortex which is capable of mapping highly nonlinear functions. CMAC has a local generalization property, obeys output superposition principle, trains quickly, and can be efficiently implemented in hardware to perform real-time mappings.

**collision:** An undesirable overlap in output space due to limitations of the hash coding procedure. Collisions can be limited by increasing weight memory size.

**convergence:** The reaching of a desired solution through a neural network iterative error-correction training procedure.

**excitation:** The process through which a field current is supplied to the magnetic circuit such that a flux is produced to link to the stator windings and induce the phase voltage.

**fault tolerant:** The ability of a component or system to continue to perform its intended function given the event of a subcomponent or subsystem failure. Specific fault tolerance of a neural network is the property that allows the system to function and gradually degrade when a small number of processing elements have been destroyed or disabled.

**flux linkage:** The product of phase current and the inductance in a magnetic circuit.

**generalization:** The ability of a neural network to generalize from the input/output examples it was trained on to produce a reasonable output from a previously unseen input.

**global learning:** A form of learning associated with backpropagation which tends to blur the details of weight updating. The learning process is analogous to fitting a low-order polynomial through a set of data points.

**gradient search:** The method of minimizing the mean squared error of the network by moving down the gradient error curve. In a simple system, the error curve is a smooth paraboloid. In this case, the network would be guaranteed to eventually reach the bottom of the curve. However, in the realistic case, there are valleys and hills (local maxima and minima) that the network must negotiate before finding the lowest point.

**hash coding:** A many-to-one mapping of the virtual address to a physical address.

**heteroassociative:** A type of neural network which requires both a training input and a corresponding desired output. This type of network expects that the input and output will be different. Backpropagation, radial basis function networks, and CMAC can all be trained as heteroassociative networks.

**inductance:** The phenomena through which a current in a magnetic circuit produces a voltage that is proportional to that current. The parameter of inductance is a proportionality constant that depends on the geometric features of the magnetic circuit.

**input quantization:** The quantization of network input values. This quantization allows a generalization of like input values.

**Insulated Gate Bipolar Transistor (IGBT):** A voltage driven field effect device which has a high current density capability and an operating frequency of up to 20 kHz.

**integral starter/generator:** An electrical machine that is integrated with a gas turbine aircraft engine. The desired mounting point for the machine is the high speed compressor shaft.

**learning rate (Beta):** A number between zero and one which is used in the weight adjustment calculation which controls the rate of network learning.

**least mean square rule:** The neural network learning rule which is also called the delta rule. The rule calculates the amount the weight should be adjusted as the product of the learning rate and the difference between the desired network output and the actual output.

**local learning:** The type of learning associated with radial basis functions and CMAC in which utilizes neurons with locally-tuned overlapping receptive fields. Local learning is analogous to fitting a least-squares spline through a set of data points using piecewise polynomials. Advantages of local learning are relatively fast learning and the ability to train in one part of the input space without corrupting another part.

**magnetomotive force:** The source of the magnetic field in the core of a magnetic circuit. This force can be expressed as the ampere turn product, the number of winding turns multiplied by the current flowing through the winding.

**Metal Oxide Semiconductor Controlled Thyristor (MCT):** The metal oxide semiconductor adds the ease of gate control to a device which is already capable of high voltages and high currents. The device has a junction temperature of 200 degrees Celsius and a switching capability of 30 kHz.

**minimum average root mean square error:** A CMAC parameter which represents the network's performance threshold at which training is ended.

**minimum delta error:** A CMAC parameter which is the minimum difference between the average rms error of the current iteration and the error of the previous iteration. When the network reaches this set minimum training is stopped.

**More Electric Airplane:** A conceptual aircraft design which would eliminate many different modes of secondary engine power extraction presently in use with an electrical generator.

**neural networks:** Information processing systems that learn using models of biological neurons. In general, neural networks can be thought of as black boxes that accept inputs and produce outputs; they perform a mapping function. Neural networks have been applied to solve many different kinds of problems including classification, optimization, prediction, controls, diagnostics, speech recognition, image processing, etc.

**number of mappings:** The number of mappings in an intermediate mapping in the CMAC scheme; the number of weights summed to compute the output.

**overlap:** The CMAC parameter used to directly adjust the amount of network generalization. The overlap is the number of slots in the CMAC look-up table which will have the same index. The overlap can also be thought of as the size of an individual neuron's receptive field.

**permanent magnet machine:** A synchronous machine in which the open circuit magnetic flux field is provided with one or more permanent magnets.

**physical address:** the address in memory in which a weight value is stored.

**pole:** A structure of magnetic material on which a field coil (winding) may be mounted.

**power density:** The ratio of machine capacity to specific weight [kW/lbs].

**Radial Basis Function (RBF):** A neural network model consisting of three layers of neurons. The first layer simply serves as the input layer; the middle layer uses a Gaussian function to model the nonlinearities of the mappings; the third layer produces the output from a summation of weighted values. The middle layer neurons have locally-tuned receptive fields.

**recall capability:** The ability of a network to map the correct outputs given input vectors with which it was previously trained.

**reluctance:** The ratio of magnetomotive force to the magnetic flux through any cross section of the magnetic circuit.

**resolution:** The CMAC parameter which determines the size of the bins created for the quantization of input values.

**rotor:** The rotating member of an electrical machine.

**starter/generator:** A machine whose intended functions include operating as a motor drive capable of starting an engine and also as a generator capable of supplying electrical power to loads.

**stator:** The stationary portion of an electrical machine. The stator includes the stationary portion of the magnetic circuit and the associated windings and leads.

**switched reluctance machine:** An electrical machine which can be operated as either a motor or a generator. The machine is controlled by switching the phase currents on and off in synchronism with the rotor.

**torque:** For a motor, the force in the direction of rotation which opposes the force of friction and is capable of driving a mechanical load. Torque is produced in a switched reluctance machine by the tendency of the nearest rotor poles to move to a minimum reluctance position with respect to the excited stator pole.

**training:** The process by which a neural network learns the relationship between the inputs and the desired output. A network is made up of a number of processing elements that can be connected in different ways. These connections are weighted and during

learning the weights are adjusted until the desired output is obtained.

**Vanadium Permendur:** A magnetic material with a relatively high permeability.

**virtual address:** The weight address for a particular input vector for a CMAC network. The address is virtual because the number of possible addresses is too large for practical implementation. The actual weight address is hash coded from the virtual address.

**weight adjustment tolerance:** The minimum difference between the computed output and the desired output that will warrant an adjustment of the weight value

**weight table:** The set of addresses in which the weight values for a CMAC network are stored.

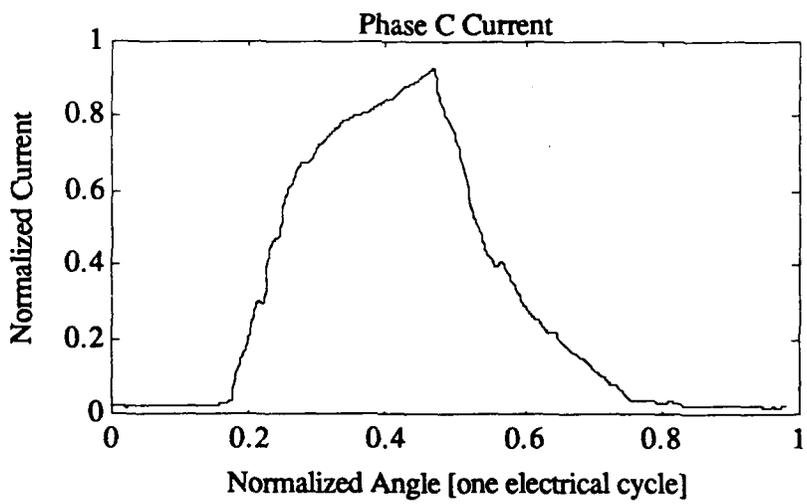
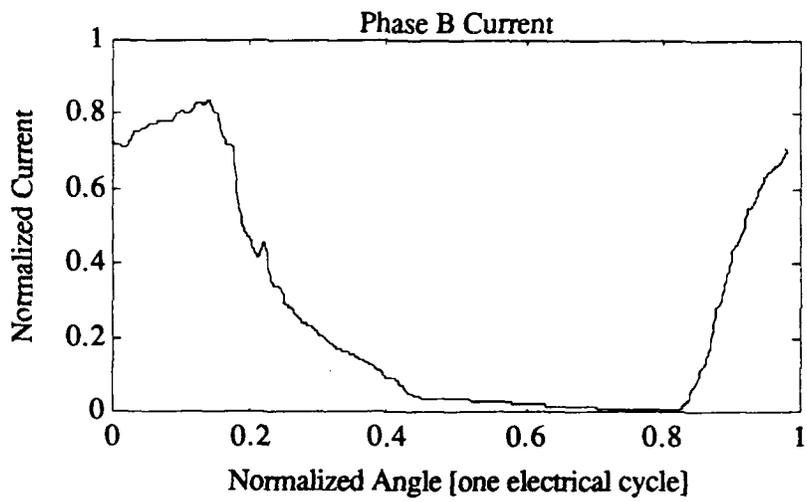
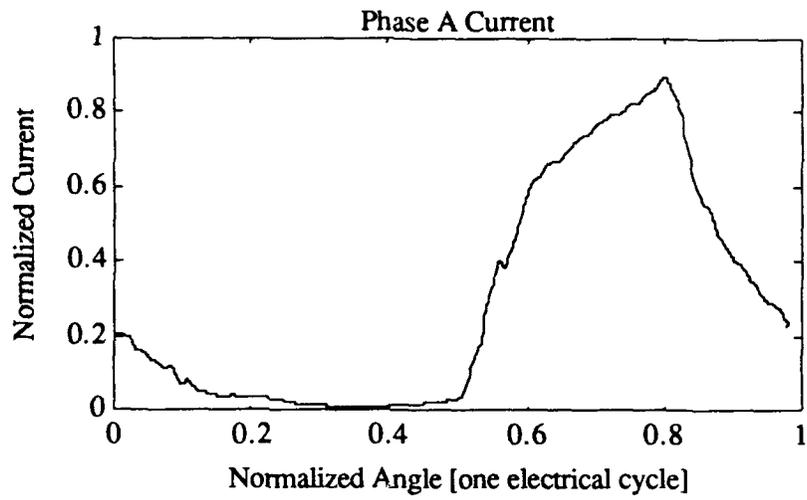
**wound rotor machine:** An electrical machine whose wound rotor carries a polyphase winding similar to, and wound for, the same number of poles as the stator.

**winding:** An assembly of coils on either the stationary or rotating part of an electrical machine whose sole purpose is the production of the electromagnetic field.

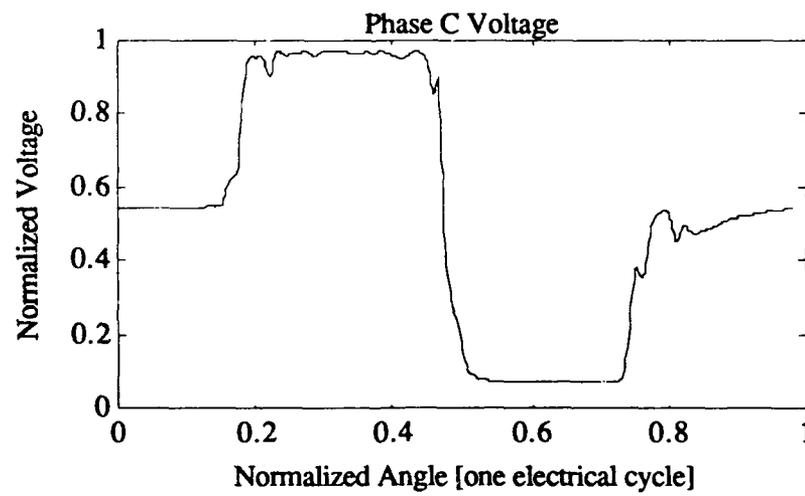
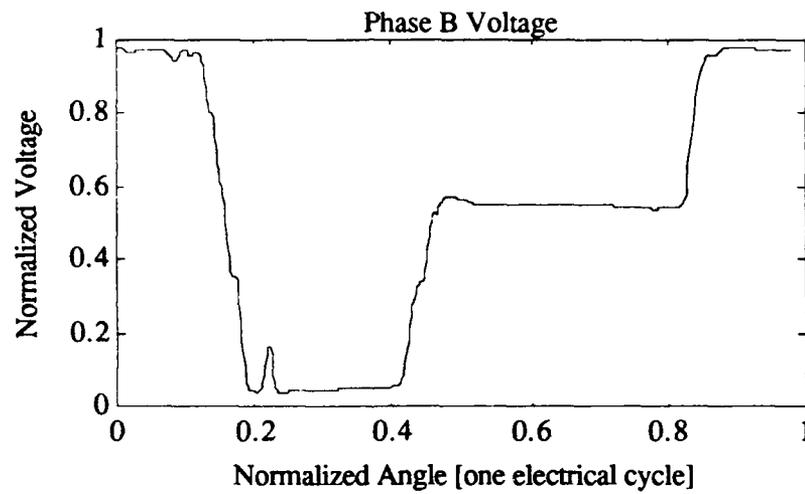
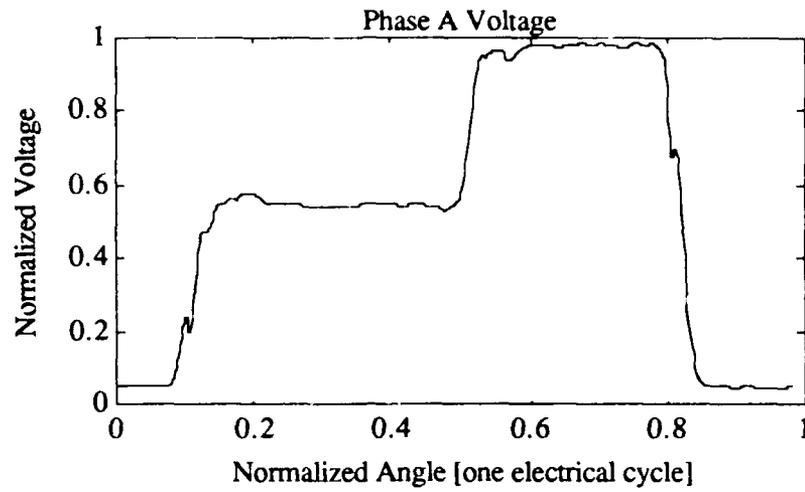
**APPENDIX B**

**FILTERED INPUTS USED FOR TRAINING**

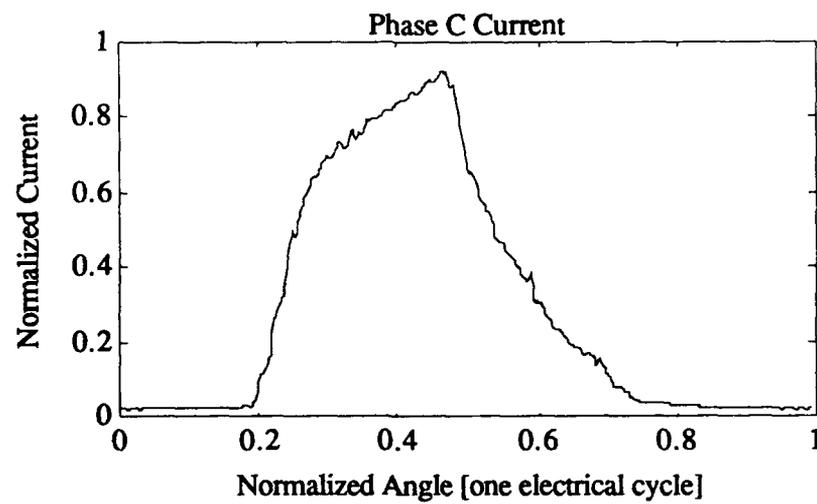
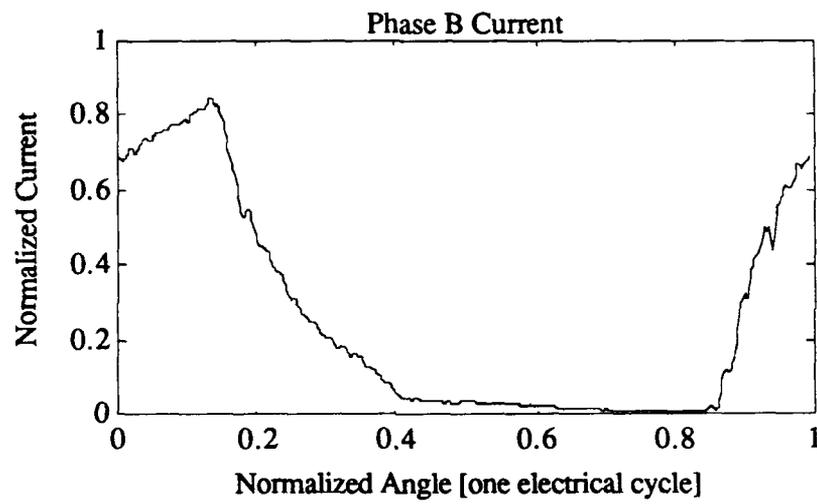
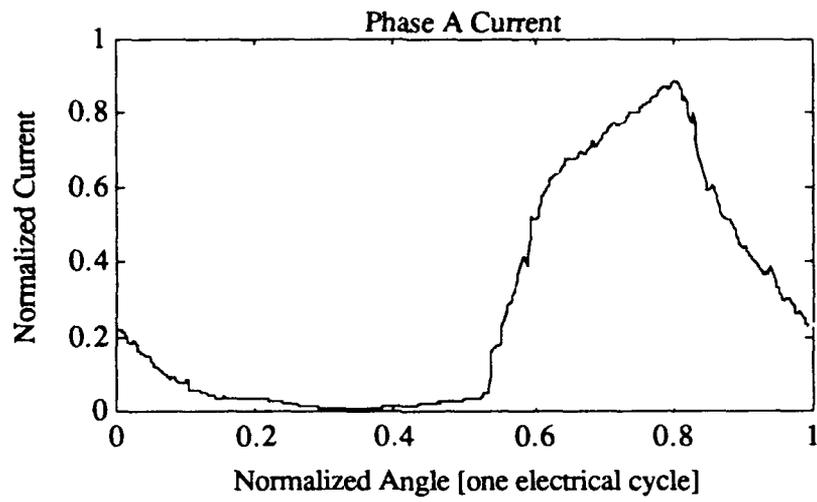
Current Waveforms from File M7k100v.300



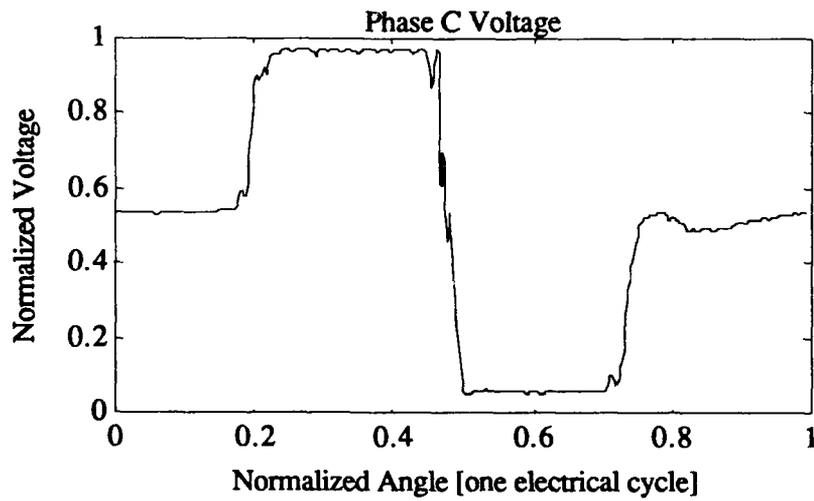
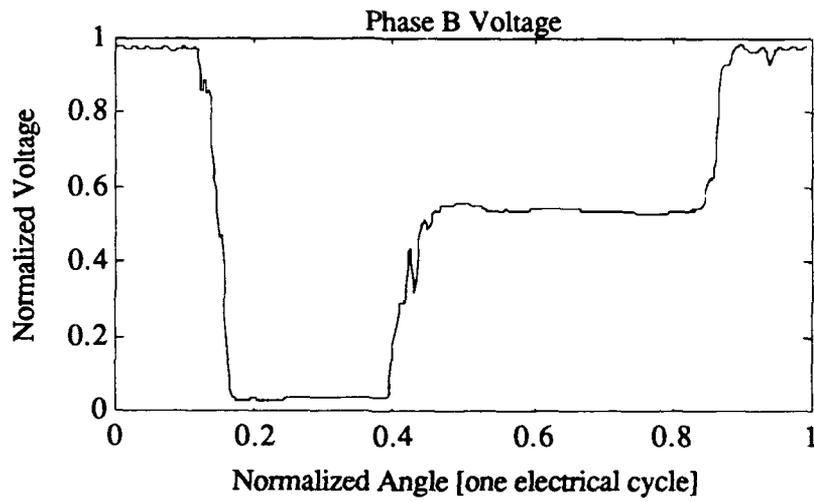
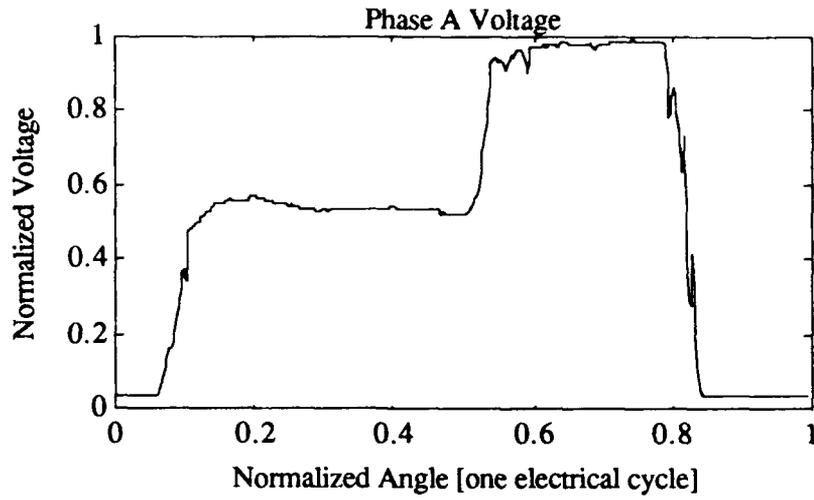
Voltage Waveforms from File M7k100v.300



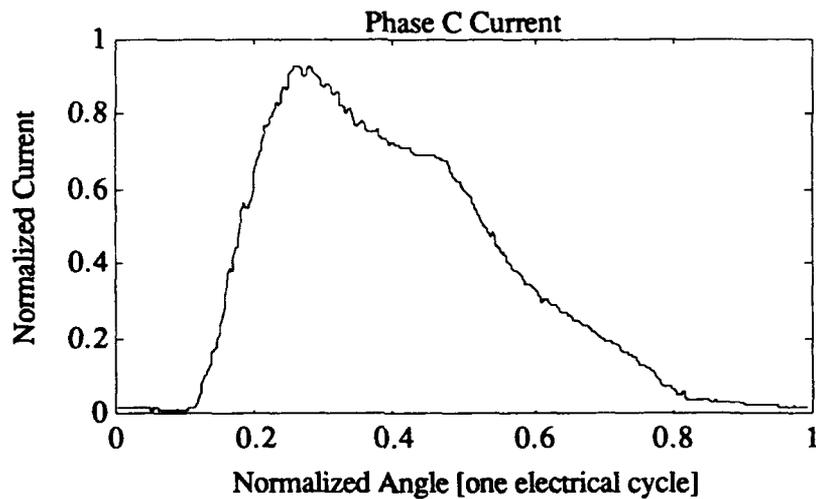
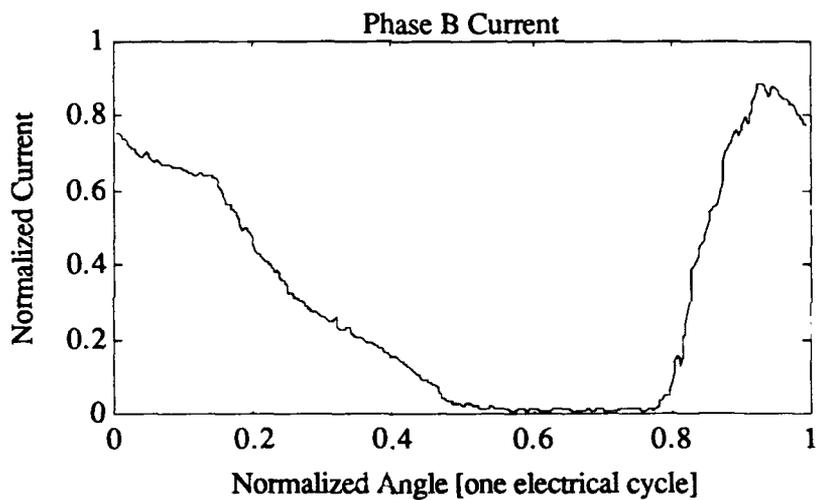
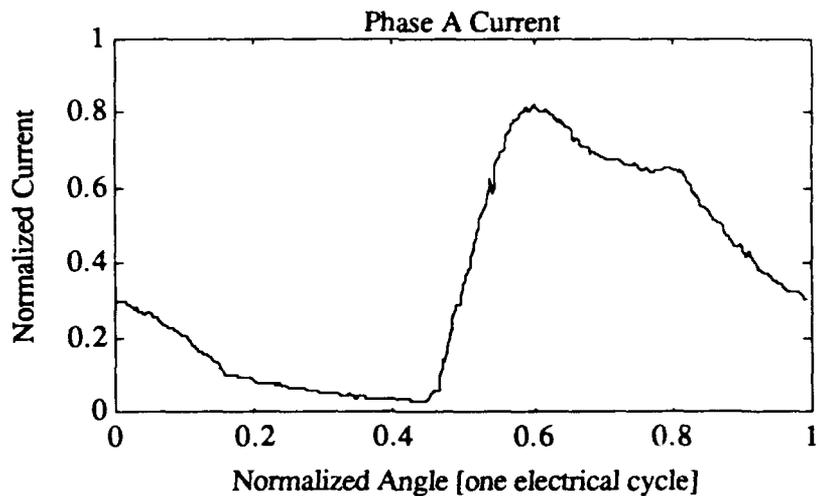
Current Waveforms from File M7k100v.225



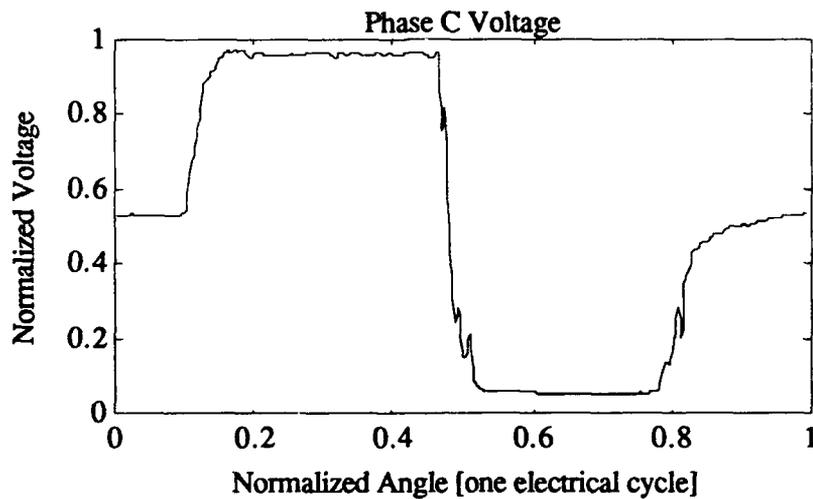
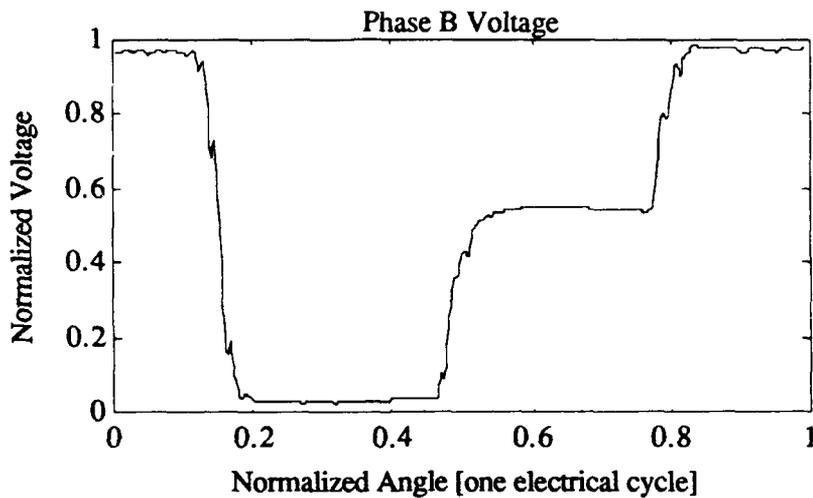
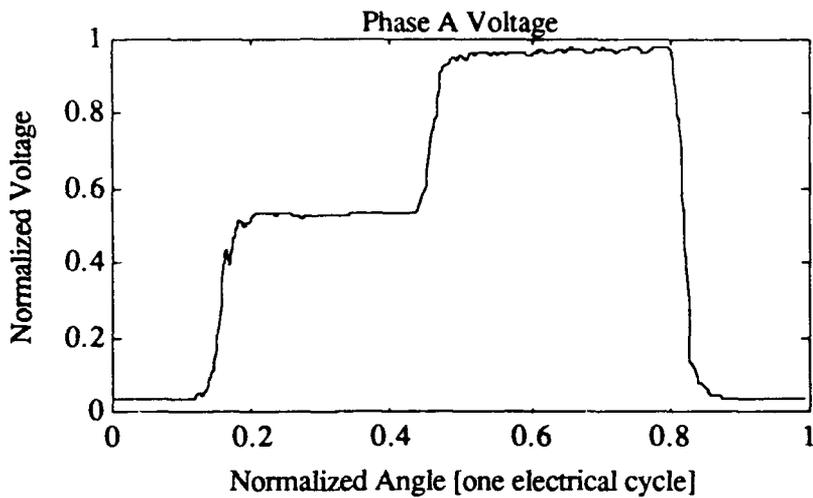
Voltage Waveforms from File M7k100v.225



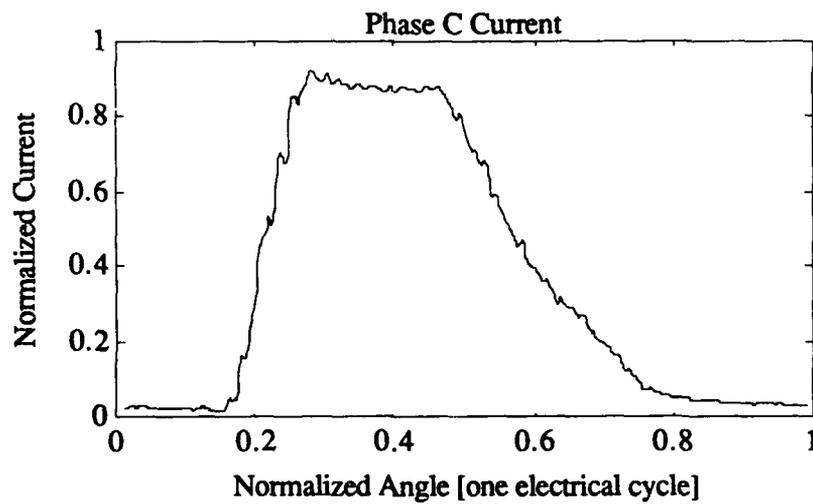
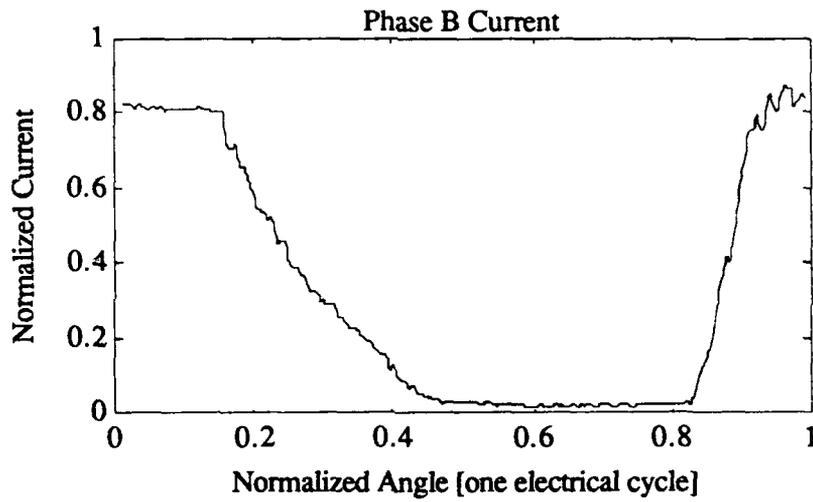
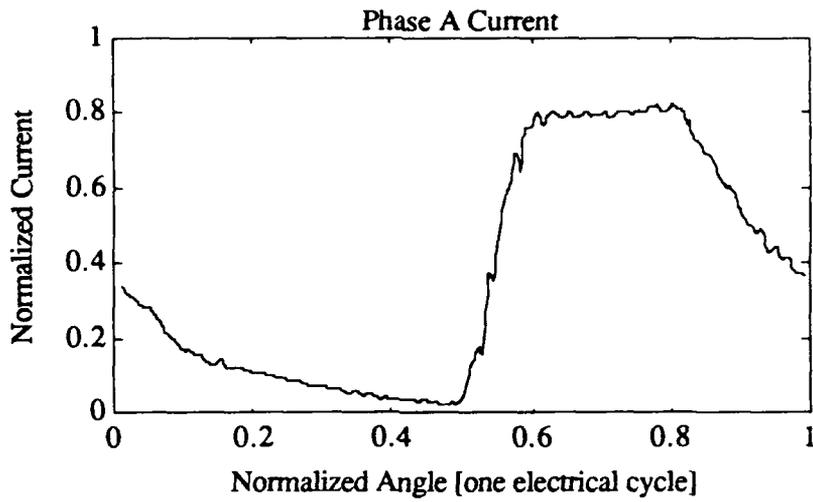
Current Waveforms from File M15k100v.300



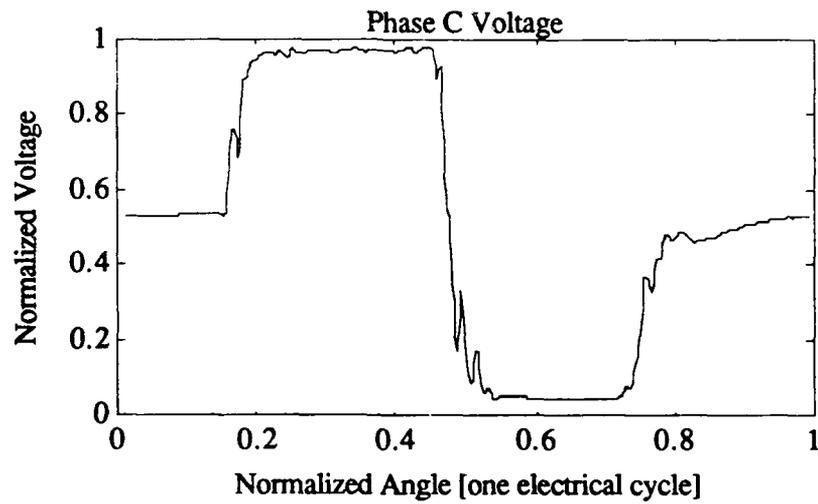
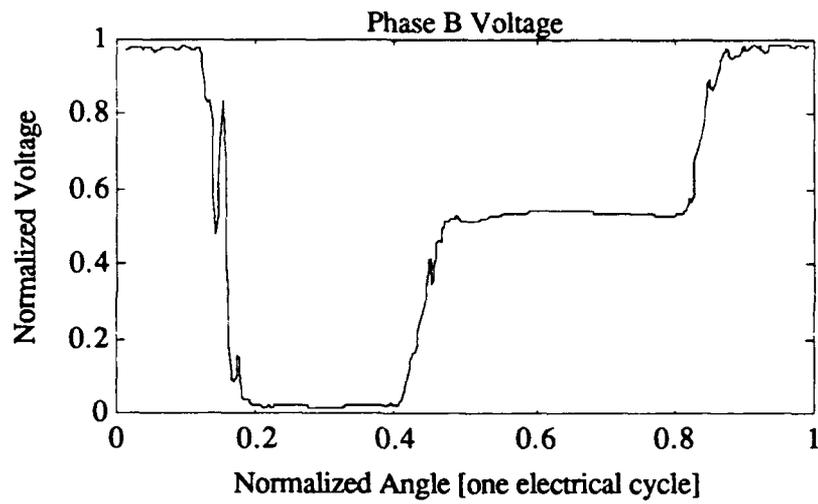
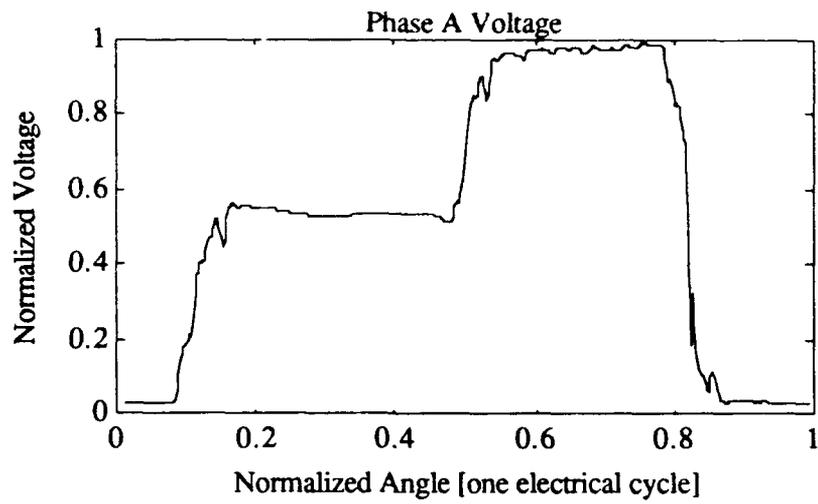
Voltage Waveforms from File M15k100v.300



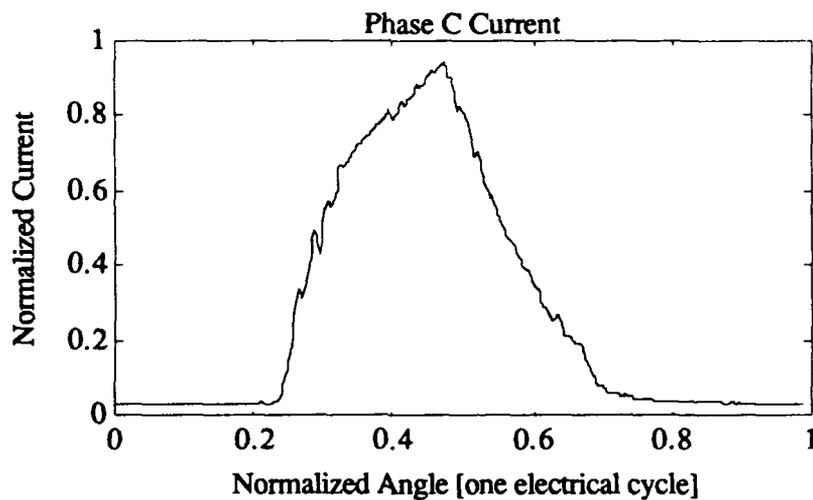
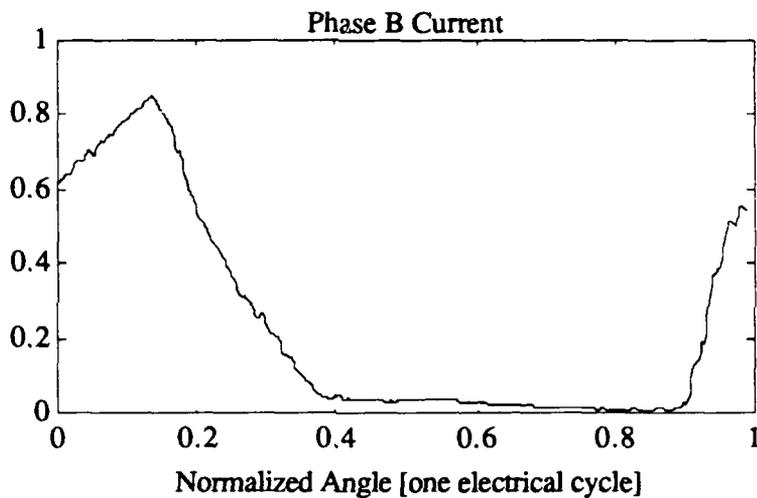
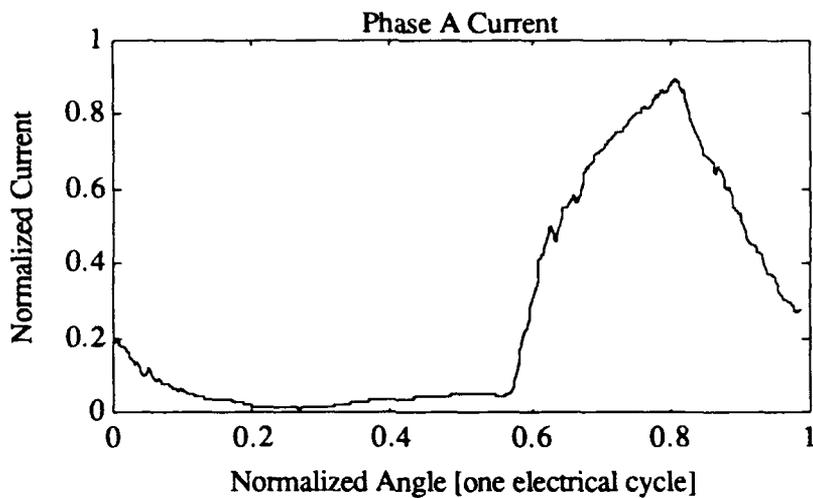
Current Waveforms from File M15k100v.200



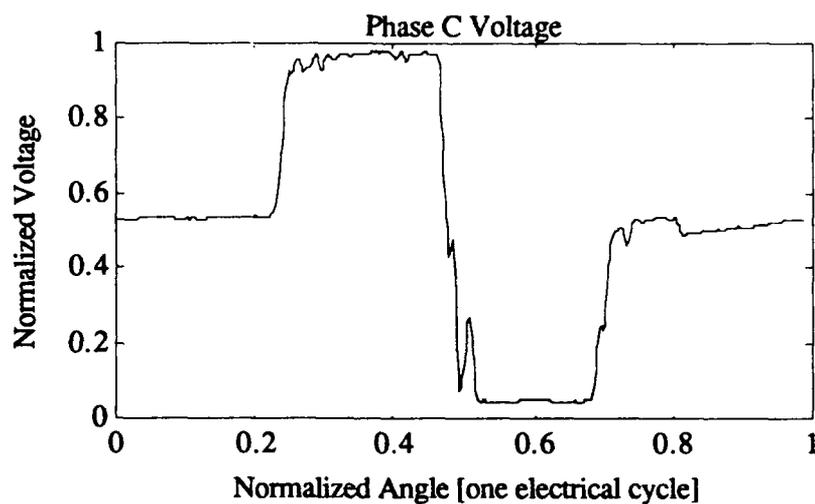
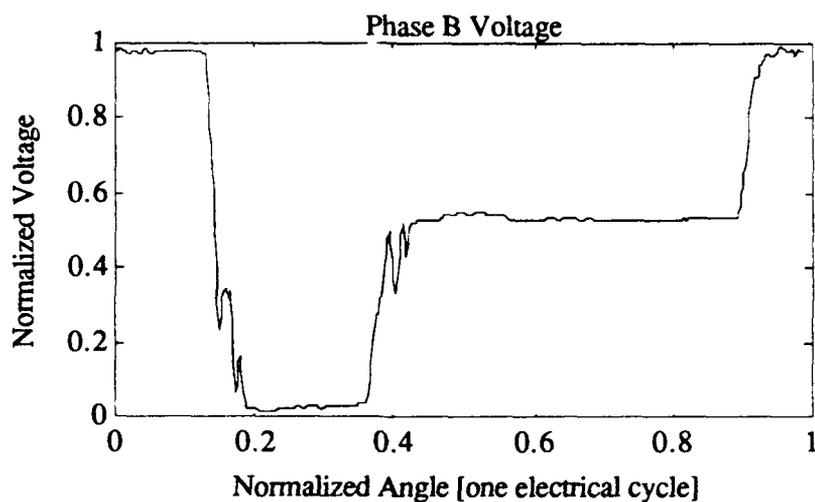
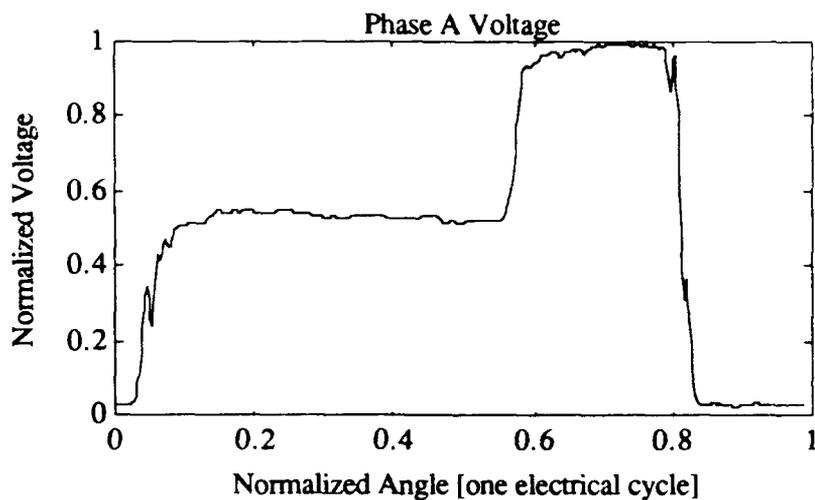
Voltage Waveforms from File M15k100v.200



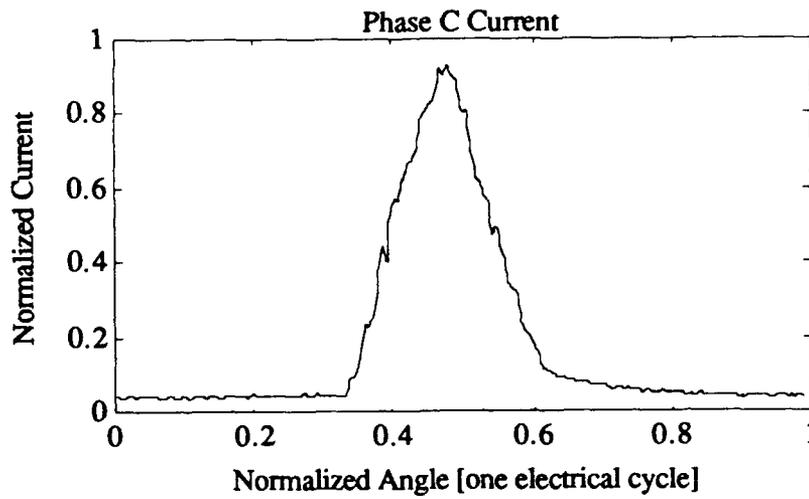
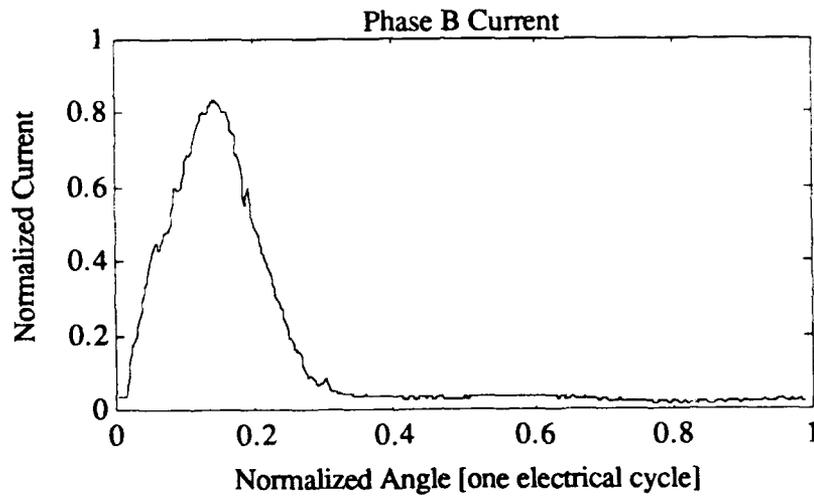
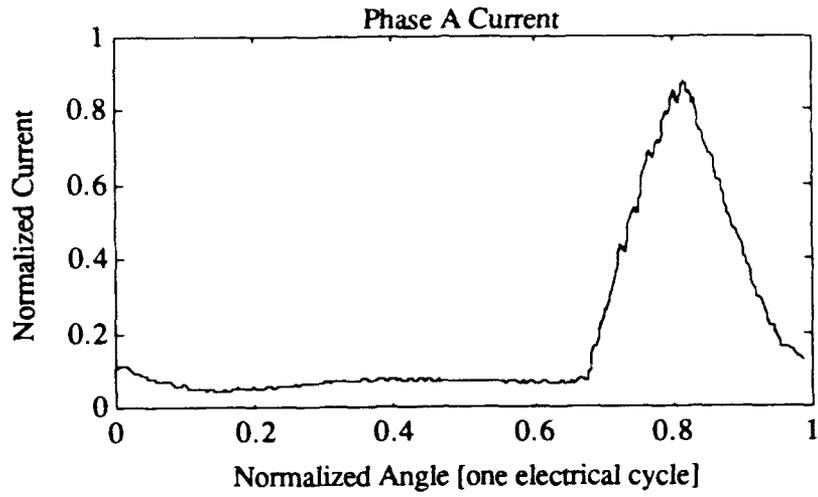
Current Waveforms from File M15k100v.150



Voltage Waveforms from File M15k100v.150



Current Waveforms for File M7k100v.075



Voltage Waveforms from File M7k100v.075

