DESIGNING ADC BY MEANS OF SOFT COMPUTING

F. SHABANINIA, E. FARJAH AND M. SAMIE

Department of Electrical Engineering
Shiraz University
Shiraz, IRAN

ABSTRACT—In this paper, a reconfigurable and learnable Analog to Digital Converter (ADC) based on soft computing is presented. It is shown that when a fuzzy system has a single input (example of an ADC), it can be multiplied in the premise of rules by introducing proper fuzzy words and linguistic operator to amplify the knowledge base. Also, in this paper, instinctive and learning behaviors, knowledge base and topology of the ADC circuit are found using a fuzzy Voronoi diagram in a POE (Phylogeny, Ontogeny and Epigenetic) model based program. This ADC implements a set of processes during the analog to digital conversion. The proposed ADC may have a more proper structure by implementing hybrid neural network because its parameters are applied in a training algorithm.

Key Words: Analog to Digital Converter (ADC), Neural Network, Radial Basis Function (RBF), Fuzzy, Soft Computing, POE Model and Voronoi

1. INTRODUCTION

The ADC and DAC converters are intrinsically interpretable by soft computing techniques [1, 2]. In a simple case, Voronoi Diagrams (VoD) introduce a feedforward neural network structure for ADC that is known as flash ADC. Since these neurons are used in hybrid neural network structures, proper neuro blocks could always be found and implemented by a set of processes during the analog to digital conversion process. This is the main idea of using neural network for designing these converters. The latest works in this area consist of using a single or mixed combination of Hopfield and feedforward neural networks, spiking neurons and neuron MOSFET [3-7]. In this paper, a reconfigurable converter based on a RBF neural network and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) network is introduced.

The basic idea in this paper is to use fuzzy Voronoi diagrams to introduce learning behaviors in a network defined algorithm based on the same Voronoi diagram (Race-Instinctive-Behavior Promotion Algorithm, RIBPA). This means that the instinctive behavior of the network that was defined by a Voronoi diagram could be improved in processing of training cycles. First, the circuit is reconfigured by increasing the number of Voronoi sites, for which the results of learning process are not responsible. Then, the learning process will be repeated approaching a minimum error. By using a fuzzy Voronoi diagram and a POE model [8], several advantages should be gained. In fact, one should benefit from the advantages of a periodic membership function: a reconfigurable circuit, a simple design, decreasing number of rules and clearer solutions. This periodic membership function can be interpreted easily with RBF neural networks or ANFIS structures. A POE program is used to enhance input effects in the premises of fuzzy rules and for circuit reconfiguration.

On the other hand, this paper presents a new strategy for the design of a reconfigurable and learnable ADC by using bio-inspiration from living beings. It seems that this new strategy can continue self-repairing the converter properties, but it is not the subject of the present work. The present paper presents a method of reconfiguration which may be superior to the learning algorithm used by others [3-7]. The sample and hold block that is the main block used in conventional ADC is completely removed. The sample and hold block reduces the speed of conversion drastically, so removing it makes the conversion process much faster. Another feature of the proposed ADC is that all of the bits could be recognized simultaneously.
In the following section the manner in which an ADC is interpreted by VoD is explained. Sections three and four are focused on expanding a fuzzy method for systems that have only a single input, such as ADC. Section four especially explains the capability of a POE based program and how to design reconfigurable and learnable circuits. The implementation of this strategy for bit identification in an ADC is presented in section five. Finally, a Neuro-fuzzy circuit for 4 and 8-bit ADC is presented.

2. THE VORONOI DIAGRAM SOLUTIONS FOR ADC

Neural network based ADC has been extended by Hopfield and others [3-7]. In this section it is shown how the neural ADC is interpretable by VoD. The neuron to neuron formation method of neural network by the Voronoi diagram [9] is based on the fact that a hyper plane equation can be implemented by a single perceptron. In accordance with A/D converter, a Voronoi diagram defines a feedforward network for it. The Voronoi sites lie in specification values of input range and produce hyper planes (Figure 1-a). These hyper planes are implemented by a collection of neurons (comparators) in the first layer. Some areas of the input range that are related to defined states of output bits may be organized by K/N digital neurons [10] in the second layer. These areas are collected in classes that extract output bits by the other layer as shown in Figure 1-b. The need for $2^n$ neurons in the first layer is a major disadvantage. A solution can be obtained by cascading the hyper planes. This strategy can be interpretable by a triangular Hopfield neural network [4] and has advantages such as: removing the necessity of areas, classes and their related layers, but it can’t recognize all bits, simultaneously (Figure 1-c).

Figure 1. a-b) Flash ADC interpreted with Voronoi diagram. c) Triangular Hopfield neural ADC.

3. FUZZY VORONOI DIAGRAM

Figure 2-a shows two sites that form a perpendicular bisector. Voronoi is a crisp line that extends infinitely in both directions, and the two half planes locate on either site. A fuzzy Voronoi diagram may be obtained by allocating fuzzy membership functions to Voronoi sites and define hyper planes by fuzzy sentences (Figure 2-b). Therefore the line will be replaced by ambiguous boundaries and could be extracted through the fuzzy inference and defuzzification. In a simple way, it is supposed that the membership function of a given site will change in passing from one site to the other one. The membership function may remain sigmoidal, Gaussian or other standard membership function shape. Fuzzy VoD diagram makes it possible to locate any position between two sites for the hyper planes boundary. Hyper plane could be described by fuzzy sentences such as those in (1):

$$H^+ \text{ is } MF_{s_1} \text{ bigger than } MF_{s_2}$$

$$H^- \text{ is } MF_{s_2} \text{ bigger than } MF_{s_1}$$

(1)

Where $H^+$ and $H^-$ are fuzzy hyper planes, $MF_{s_i}$ are membership functions of Voronoi sites, and “bigger than” is a fuzzy operator.

4. MULTIPLICATION OF INPUT VARIABLE

In an application such as ADC, where there is a single input to the fuzzy system, it is possible to improve the weakness of the fuzzy system which is due to low input linguistic variables by multiplication of input variables in the premise of rules. This multiplication creates a mapping from the initial space to the space with more dimensions of patterns, which produces more knowledge for fuzzy inference system. The fuzzy Voronoi diagram inserts the fuzzy words and sentences in the premise of rules, so that may be considered as an appropriate strategy for this purpose.
On the other hand, Fuzzy VoD diagrams provide mechanisms for testing and appraising a unique problem in different sets of patterns. Hence, it obtains more complete information for fuzzy inferences. Hyper planes and knowledge of fuzzy VoD diagram can be represented by fuzzy sentences and it is possible to insert them into the measuring of rules, words, linguistic variables and rules structures that are found by using genetic programming, Figure 3-a. Consider the following fuzzy rule that has $n$ multiples of each linguistic variable:

$$\text{if } \prod_{j=1}^{m} \prod_{i=1}^{n} [x_{ji} \cdot (\text{Verb}_{ji}) \cdot A_{ji}] \cdot [\text{Conjunction}_{ji}] \cdot [\text{Averb}_{ji}] \cdot [\text{Adjective}_{ji}] \text{ Then } y \text{ is } B$$  \hspace{1cm} (2)$$

$x_{ji}$ is $i_{th}$ multiplication of $j_{th}$ input variable, $A_{ji}$ is $j_{th}$ membership function for $i_{th}$ multiplication of $j_{th}$ input variable, $y$ is output linguistic variable, $B$ is output membership function, $n$ is the number of multiplication, $m$ is number of input variables, Verb, Conjunction, Adverb and Adjective are fuzzy words from following four fuzzy sets:

- Verb = \{is, isn’t, non\}
- Conjunction = \{and, or, non\}
- Adverb = \{Very, more and less, non, etc.\}
- Adjective = \{Bigger than, Smaller than, non, etc.\}

In accordance with the algorithm in Figure 3-a, fuzzy rule 2 indicates that $n$ is the number of fuzzy Voronoi sites.

This algorithm is a Race-Instinctive-Behavior Promotion process based on POE model inspired by biological systems when applied to chromosomes with a floating length. Data set sections of the chromosome are divided into noncoding and coding determined information content (number of centers-location of centers VoD). These sets of data introduce a raw network for which the natural treatment reformation is to be done by a cyclic learning due to an information site; this cyclic learning is applied to each of the data sections.

This means that the topology or shape of a circuit will determine a phylogeny step so that it will define a raw network. It will then improve in an epigenetic step to set the synapses of neurons. The shape will change if learning iteration does not fall below a minimum error. So the circuit will execute behavior of a reconfigurable neuron (Figure 3-b). It is possible to continue an ontogenetically step to attach self-repairing properties for embryonic strategies.

Among the existing topologies, the RBF [11] neural network may be considered as a practical guideline for implementing fuzzy rule 2, since it could be interpreted by an ANFIS neuro-fuzzy network [12]. An ANFIS network could sustain the inference system by learning more about the problem and expanding the space. Particularly, the circuit will be reconfigured, when it changes the number of fuzzy Voronoi sites in fuzzy rule 2. On the other hand, it is observed that the shape or topology of the circuit has been changed.
5. BIT IDENTIFICATION BY ANFIS NETWORK

An ADC converter can be considered as a single input, multiple output system. According to the previous discussion, an ANFIS network could be used for determining of the output digital bits. The network parameters are identified by the instructive software. In this work, MATLAB 6.1 fuzzy toolbox is used. By considering four Gaussian membership functions and 350 iterations, rules and input membership functions for a bit of significance $2^1$ could be obtained. One should note that these membership functions are pseudo periodic.

Otherwise, a VoD fuzzy diagram is used in a program modeled by POE [13] and the algorithm in Figure 3 for the input linguistic variable distribution. Then, the membership functions and rules will be obtained for any chromosome which has small error. Combining the basis of the recent membership functions and rules results in a special periodic membership function; these rules can be simplified to give the following rule (rule 3).

$$
\text{if } x \text{ is } A \text{ bigger than } x \text{ isn't } A \text{ then } D_j \text{ is '1'}
$$

Where $x$ is input variable, $D$ is output in Sugeno inference, bigger than and isn't, are fuzzy operators, and $A$ is a periodic membership function for a bit of significance $2^1$, Figure 4-a.

![Figure 4. a) periodic membership function for a bit of significance $2^1$, b) Chromosome schematic.](image)
information of the content of chromosomes should be saved in proper registers (Figure 3-b), and then applied to switches in Figure 5.

6. PROPOSED HARDWARE FOR BIT IDENTIFICATION

The proposed strategy could be practically implemented using the Neuro-Fuzzy Architecture for CMOS Implementation presented in [14]. According to this architecture, a periodic membership function could be obtained by repetition of difference stages and may also be considered as a neuron based on RBF neural network. In this work, the architecture presented in [14] is improved by adding synapses that are reconfigurable and learnable. A Resultant Neuron circuit is shown in Figure 5 for a three site fuzzy Voronoi diagram. V_{rq} voltages give the transition points through the medium range variations of input-output relation in an instinctive behavior. The state of P_{hi} switches will change if the number of Voronoi sites changes in a phylogeny step. Changing the V_{rq} and the switches labeled E_{pi} is possible in a learning process (Epigenetic).

Results for a double periodic activation function are shown in Figure 6. Each neuron not only generates the desired output but also gives its complement. This is an advantage of the above architecture in relation to a simple implementation of rule 3. The membership function is defined by the output current I_i in medium range variations will be represented by the following formula:

\[
I_i = I_{bias} \left[ 1 + 0.5 \left( \frac{\alpha_i}{\sqrt{2} \beta_i - \alpha_i^2} - \alpha_{i+1} \sqrt{2 \beta_{i+1} - \alpha_{i+1}^2} \right) \right]
\]

in which:

\[
\alpha_i = (V_{IN} - V_{REF})/V_{TH}
\]

\[
\beta_i = K W_i V_{TH} / 2 L_i I_{bias}
\]

7. 4-BIT NEURO-FUZZY ANALOG TO DIGITAL CONVERTER

The 4-bit neuro-fuzzy analog to digital converter could be viewed as a single input-multiple output ANFIS network for which the internal structure may be a combination of RBF neural networks for generating periodic membership functions (Figure 7). Every RBF neural network is based on the block diagram of Figure 3-b and the circuit shown in Figure 5. These neurons can represent a fuzzy concept for the input signal by applying the activation function with periodicity 1, 2, 4 and 8. Each neuron also
generates the complementary fuzzy concept of the input signal. Finally, the Sugeno inference shown below would extract the output bits directly.

\[
\begin{align*}
R1: & \quad \text{if } (x \text{ is small}) \text{ bigger than } (x \text{ isn't small}) \quad \text{Then } D_0 \text{ is '1'} \\
R2: & \quad \text{if } (x \text{ is medium 1}) \text{ bigger than } (x \text{ isn't medium 1}) \quad \text{Then } D_0 \text{ is '1'} \\
R3: & \quad \text{if } (x \text{ is medium 2}) \text{ bigger than } (x \text{ isn't medium 2}) \quad \text{Then } D_0 \text{ is '1'} \\
R4: & \quad \text{if } (x \text{ is big}) \text{ bigger than } (x \text{ isn't big}) \quad \text{Then } D_0 \text{ is '1'}
\end{align*}
\]

Where \( x \) is the input variable, \( D_i (i=1, 2, 3, 4) \) is the output in the Sugeno inference for the bits by significant of \( 2^i \), bigger than and isn't, are fuzzy operators. Finally, small, medium1, medium2 and big are periodic membership function by \( 2^i \) period, respectively.

In order to show the feasibility of the method, the ADC is tested with different input signal. Figure 8 shows the simulation results for a simple sinusoidal waveform of frequency 200 kHz and a peak to peak value of 5 V as the input to the ADC. The generated least significant bit (LSB) and most significant bit (MSB) are also shown in the same figure. A good accordance is seen when comparing the input signal with that generated by identified bits (inverse D/A).

**8. 8-BIT NEURO-FUZZY ADC**

In order to increase the resolution, one should use membership functions with numerous periods, but this leads to a large size of the circuit and problems due to threshold voltage regulation may cause it to be impractical. This problem can be solved using a fuzzy folding technique (Figure 9). To determine the rules which deduce the outputs of a primary layer, we use the distribution of linguistic variable in the premise of
rules. By following the design strategy of the previous section, the complementary rules will be added to the rules of previous sections as below:

\[ R5: \text{if } (x \text{ is } mf1) \text{ then } y2 \text{ is } mfo2 \]
\[ R6: \text{if } ((x \text{ is } mf1) \text{ and bigger than } (x \text{ is } mf2)) \text{ or } ((x \text{ is } mf1) \text{ and bigger than } (x \text{ is } mf4)) \text{ then } y2 \text{ is } mfo2 \]
\[ R7: \text{if } ((x \text{ is } mf2) \text{ and bigger than } (x \text{ is } mf1)) \text{ or } ((x \text{ is } mf2) \text{ and bigger than } (x \text{ is } mf3)) \text{ then } y2 \text{ is } mfo3 \]
\[ R8: \text{if } ((x \text{ is } mf3) \text{ and bigger than } (x \text{ is } mf2)) \text{ or } ((x \text{ is } mf3) \text{ and bigger than } (x \text{ is } mf4)) \text{ then } y2 \text{ is } mfo4 \]
\[ R9: \text{if } ((x \text{ is } mf4) \text{ and bigger than } (x \text{ is } mf1)) \text{ or } ((x \text{ is } mf4) \text{ and bigger than } (x \text{ is } mf3)) \text{ then } y2 \text{ is } mfo1 \]

Membership functions are similarly obtained for both input and output linguistic variables (Figure 10).

This leads to some simplifications in the inference motor and converter architecture. For example in R6 rule, the reformed rule can be written as:

\[ R6: \text{if } ((x \text{ is } mf1) \text{ and bigger than } (x \text{ is } mf2)) \text{ or } ((x \text{ is } mf1) \text{ and bigger than } (x \text{ is } mf4)) \text{ then } y2 \text{ is } mf2 \]

Where x is the input variable, \( y_i \) (i=1, 2) are the outputs of the fuzzy block in Figure 9, bigger than and isn't, are fuzzy operators. Finally, \( mf, mf_1 \) and \( mfo_1 \) (j=1, 2, 3, 4) are periodic membership function by 2^i period, respectively (Figure 10).

Therefore, the first stage of the three RBFNN is established for implementation of 2 neurons of periodicity 4 and one neuron of periodicity 1. For practical implementation, one should consider some modifications in a previously proposed architecture. The simplified network that gives out \( y_2 \) is shown in Figure 11. M1 and M2 transistors perform an OR (MAX) function and at the same time the conjunction
In order to locate $y_1$ & $y_2$ in a proper interval, shifting and amplification are necessary. This architecture needs just nine rules.

In order to show the feasibility of the method, a ramp input signal with a frequency of 100 kHz is selected. Simulated results are shown in Figure 12. It is to be noted that spikes in the generated analog signal should be considered as a penalty in the learning process. So, in order to reduce these spikes, one should continue the learning process. Simulation results verified this hypothesis and it is noted that after adequate learning and by setting proper $V_{ri}$ (Figure 5) in an epigenetic step, the spikes have been totally removed.

9. CONCLUSION

It was shown that by using soft computing techniques, a reconfigurable and learnable ADC could be designed. It is noted that the weaknesses of fuzzy logic in processing systems that have few inputs (case of ADC) could be improved by inserting the fuzzy words and operators in premise of rules to multiply the input by a fuzzy Voronoi diagram. This procedure would lead to the application of periodic membership functions, thus resulting in advantages such as decrease of membership functions and rules, transparency of rules, simple inference motor and finally an integrated circuit architecture.

Hardware POE model inspired from living organism shows that signal conversion action is due to internal behavior of the converter. Reconfiguring ability and simultaneous bits identification are most advantageous compared with other works in the literature. The proposed new strategy is capable of being as fast as other ADC’s. These properties are added to hardware especially from fuzzy Voronoi diagram. Examples given for 4-bit and 8-bit ADC show the feasibility of the proposed method.

REFERENCES


ABOUT THE AUTHORS

Dr. F. Shabaninia received the B. S. Degree in EE from University of Washington, USA, and the M. S. Degree in EE from San Jose State University, USA, and PhD Degree in EE from New Mexico State University, USA, in 1983, 1985, and 1996, respectively. He taken every opportunity to teach and research in various educational university environments in USA. His research work is in intelligent control systems using soft computing methodologies. He is currently assistant professor in EE department of Shiraz University.

Dr E. Farjah obtained his B. Sc. Degree in Electrical and Electronics engineering from Shiraz University, Shiraz-Iran in 1987 and his M. Sc. Degree in Electrical Power engineering from Sharif University of Technology in 1989, Tehran-Iran and his Ph.D. in electrical Engineering from INPG, Grenoble-France. He is currently assistant professor in EE department of Shiraz University. His research field includes soft computing techniques applied to electronics and power Electronics design.

Mr. M. Samie obtained his B.Sc. in Electronics Engineering from Saveh University-Iran in 1998 and his M. Sc. Degree in Electronics engineering from Shiraz University in 2002. He is currently a senior research engineer in Tide Water Company in Iran. His research of interest includes soft computing techniques and embryonic in the field of electronics design.