STM JOURNALS

Modeling of 8-bit Logarithmic Analog to Digital Converter Using Artificial Neural Network in MATLAB

Pooja S. Shukla[1]*, Archana B. Yadav[1], Raksha K. Patel[2]
[1]Asst. Prof., Instrumentation & Control Engineering Department, SVIT, Vasad, Anand, Gujarat, India
[2]Asst. Prof., Biomedical Engineering Department, Ganpat University, Mehsana, Gujarat, India

ABSTRACT

An artificial neural network (ANN), usually called "neural network" (NN), is a mathematical or computational model that tries to simulate the structure and/or functional aspects of biological neural networks. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. Modern neural networks are non-linear statistical data modeling tools. They are usually used to model complex relationships between inputs and outputs or to find patterns in data. The logarithmic analog to digital converter gives logarithmic digital output of the given analog input signal. Basically, it is used to increase input dynamic range. Logarithmic ADC also provides a non-uniform quantization, thus compressing the input signal to digital. This is widely used in communications, instrumentation, medical instruments like deep brain stimulation, etc. In this paper, we have tried to have logarithmic analog to digital converter with neural network in MATLAB using a perceptron design.

Keywords: *Artificial neural network (ANN), perceptron, logarithmic analog to digital converter*

****Author for correspondence*** E-mail: poojashukla20@yahoo.com

1. INTRODUCTION

Logarithmic analog to digital converter (ADC) is a combination of logarithmic conversion and analog to digital conversion. This can be achieved easily with microcontrollers or with many combinations of different types of ADCs and logarithmic amplifier with operational amplifier and level shifter if needed. It gives digital logarithmic output of the given input analog signal.

Here an attempt is made to accomplish this function with analog to digital conversion only without logarithmic amplifier, level shifter circuits and microcontroller. The ADC itself will give the logarithmic digital output instead of just digital output of a given input analog signal. So, the single board will represent Log ADC. We also succeeded in making an 8-bit logarithmic ADC, but the output on the CRO was inverted from what we were getting theoretically. So, we have tried to model the same Log ADC with neural network in MATLAB with the readings what we get practically from Log ADC.

An artificial neural network (ANN) (Figure 1) is an information processing paradigm. It is inspired by the way biological nervous system, i.e., the brain processes information. The key element of artificial neural network is the novel structure of the information processing system. It consists of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. Like people, ANNs learn by example. ANN is configured for a specific application, such as

pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between neurons. This is true of ANN as well [1–5].

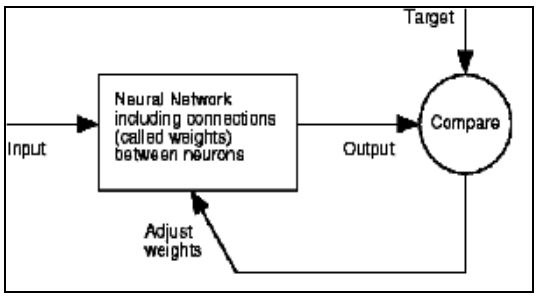

Fig. 1: *Simple Neural Network.*

Logarithmic analog to digital converter (Log ADC) is used to increase the input dynamic range. It is basically used in biomedical applications like deep brain stimulation. Generally, each signal that is taken from the body (say EEG) will always have low amplitude and higher components of noise; so, if we pass it through logarithmic ADC it will amplify the area of interest signal that is EEG signal and suppress the noise component and simultaneously convert it to digital [3].

2. DESIGN OF ANALOG TO DIGITAL CONVERTER

Here, we have considered tracking ADC because of its ability to continuously track the analog signal. The tracking ADC is one of a number of techniques which employ a DAC in a negative feedback loop. It is a type of counter-type analog to digital conversion. When the DAC output is below the analog input, the up/down input is high and the counter counts up. When the DAC output is over the analog input, the up/down input is low and the counter counts down alternatively [2].

The block diagram of a tracking ADC is shown in Figure 2 [1].

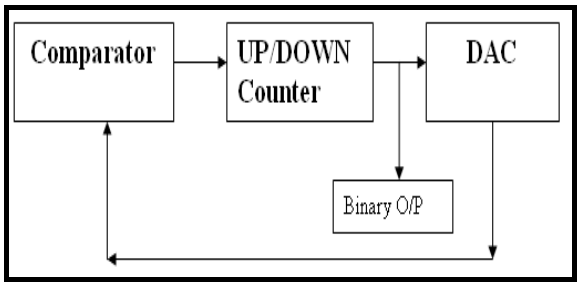

Fig. 2: *Block Diagram of Tracking ADC [1].*

The tracking ADC continually compares the input signal with a reconstructed representation of the input signal. The up/down counter is controlled by the comparator output. If the analog input exceeds the DAC output, the counter counts up until they are equal. If the DAC output exceeds the analog input, the counter counts down until they are equal. It is evident that if the analog input changes slowly, the counter will follow, and the digital output will remain close to its correct value. If the analog input suddenly undergoes a large step change, it will be many hundreds or thousands of clock cycles before the output is again valid. The tracking ADC therefore responds quickly to a slowly changing signal, but slowly to a quickly changing one [6–11].

Here, we have the design for 8-bit ADC with input range from 0 to 10 V.

The comparator block consists of two comparators; one is for completing the feedback loop with the output of DAC. As this is an 8-bit ADC with input range 0 to 10 V, the minimum step size will be $10/2^8$, so it will be 0.039 V. The minimum difference between input and f/b path will be 0.039 V with differential amplifier gain 68 as $V_{out} = R_f/R_1 (Vin_2 – Vin_1)$, where R_f = 680 k and R_1 = 10 k. And its output is given to another comparator which is for giving the up/down counting pulse for the comparators. As the first comparator output will be minimum 0.039 * 4 = 2.5 V, for the upper comparator we have set 4.8 V range and for the down comparator it is 300 mV. The upper comparator will operate only when the input is below 4.8 V and the down comparator will operate only when the input is above 300 mV. So, both comparators will give logic 1 output between 0 and 5 V. Voltage divider circuit is used for making upper comparator value at 4.8 V and down comparator value at 300 mV. The down comparator is used as up/down bit (pin No. 10 of 4029 IC) as we want input range in 0 to + 10 V. If we want the input range from −10 V to 0 V, the upper comparator may be used for the same. LM124 is used for both the comparators; one can choose any opamp IC.

The up/down counter will give binary output bits D0 to D4. Here, we have used 4029 CMOS counter IC, which can count up or down, and in either decimal or binary mode.

Digital to analog converter is used to convert digital signal given by up/down counter to analog for the feedback path. This reconstructed analog signal is continuously compared with the original analog signal. And difference between this to signal is always 0.039 V for getting 8-bit ADC response. Here, DAC is operated in unipolar mode for getting 0 to 10 V input range.

3. PROCEDURE TO CONVERT ADC TO LOG ADC

The procedure of converting the above tracking ADC to Log ADC will be clearer by block diagram shown in Figure 3.

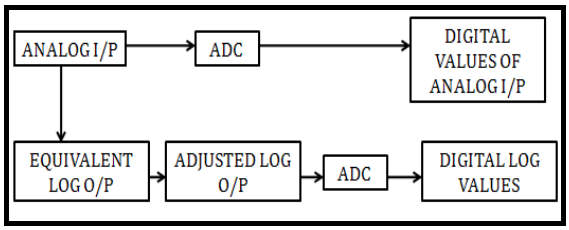

Fig. 3: *Procedure to Convert ADC to Log ADC.*

So, now we have to have some medium that gives digital log values for given digital input values. This task is accomplished by using EPROM circuit in read mode only

3.1. Design of EPROM Circuit for Logarithmic Response

Here, we have used 2764 EPROM. The 27C64 is a high-speed 64K UV erasable and

electrically reprogrammable CMOS EPROM, ideally suited for applications where fast, pattern experimentation and low power consumption are important requirements. The 27C64 is designed to operate with a single 5 V power supply with 10% tolerance [9]. The 27C64 is packaged in a 28-pin dual-in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written electrically into the device by following the programming procedure. We have programmed the EPROM 2764 with DYNALOG kit.

Here, A0 to A12 are address lines, which are connected with ADC digital output pins. As we are using 8 bit, only pins A0 to A7 are connected with ADC digital output lines and A8 to A12 pins are grounded. And O0 to O7 are output pins which are connected to DAC as feedback for logarithmic ADC response instead of ADC output to DAC [9].

EPROMs are used in different modes for different operation which are:

- Read mode
- Write mode
- Program mode

The timing diagram for read mode from datasheet is shown in Figure 4.

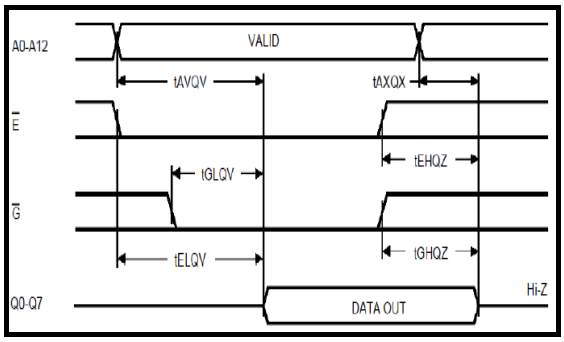

Fig. 4: *Timing Diagram of Read Mode for 2764 EPROM [9].*

In our application, we do not need to use it in the write mode; once it is programmed, we use it in the read mode only. So, the chip-enable pin is at logic 0, the output-enable pin is also at logic 0 and we are not programming it once it is programmed, so this pin is at logic 1 state.

3.2. Design of Logarithmic Analog to Digital Converter

The modified block diagram of tracking ADC for the log digital output is shown in Figure 5.

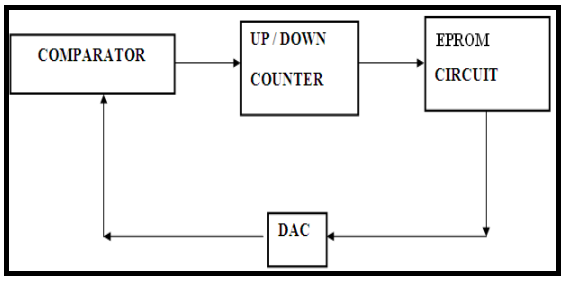

Fig. 5: *Block Diagram of Logarithmic ADC.*

The log output by realizing the above block diagram is shown in Figure 6.

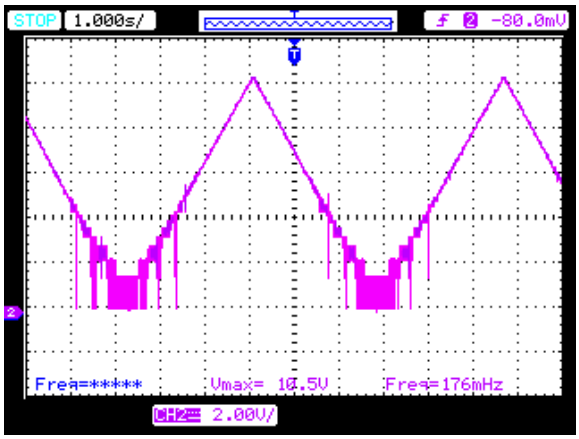

Fig. 6: *Log Digital Output for Given Analog Signal.*

This output is the compression of 255 digital steps to 125 steps. This output is digital representation of analog output what we are getting from simple logarithmic amplifier for same 0 to 10 V input as shown in Figure 7.

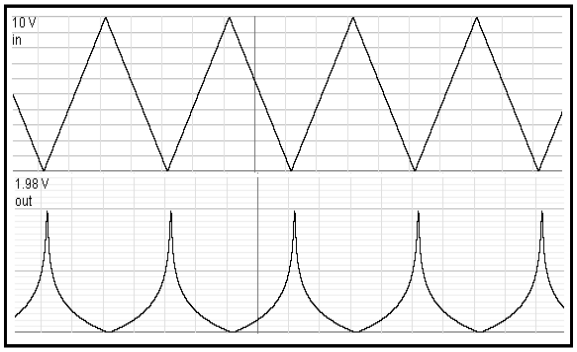

Fig. 7: *Logarithmic Output For + 10 V Triangle Wave Input* [10].

As we have considered input range positive (0 to + 10 V) we get this output. If we consider negative input range 0 to −10 V then we will get inverted output but theoretically we consider that the logarithm of ramp signal is parabolic. But it is the electronic property of the hardware to have such inverted output; so, to verify this, we have implemented the same with neural network.

3.3. Why We Use Neural Networks?

Neural networks have remarkable ability to derive meaning from complicated or imprecise data; they can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer "what if" questions. Other advantages include [12]:

a. Real-time operation: Using special hardware devices, ANN computations may be carried out in parallel.
b. Adaptive learning: ANN has the ability to learn, how to do tasks based on the data given for training or initial experience.
c. Fault tolerance via redundant information coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.
d. Self-organization: ANN has the capability to create its own organization or representation of the information it receives during learning time [12].

4. ANALYSIS AND RESULTS

In the present analysis, we have written code for analog to digital converter in MATLAB.m file to train the neural network. The data for input is given as 0 to 255 for 8 bit. And the output data is the log digital value of the input. These values are taken from the previously implemented hardware for Log ADC (Figures 8–12). This analysis was just to ensure that the implemented Log ADC gives logarithmic output; the inversion of signal is just due to the electronic components' characteristics. Here, feed-forward perceptron network is used.

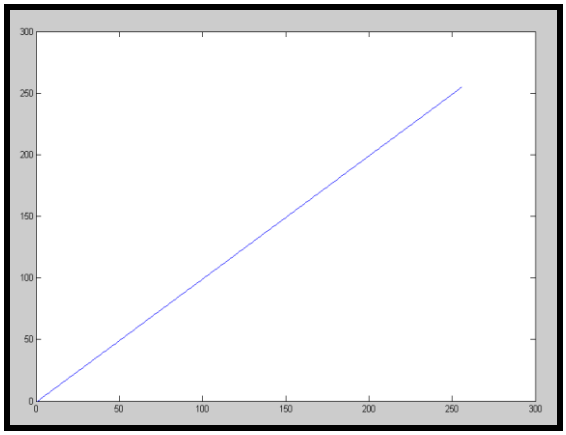

Fig. 8: *Input to Neural Network.*

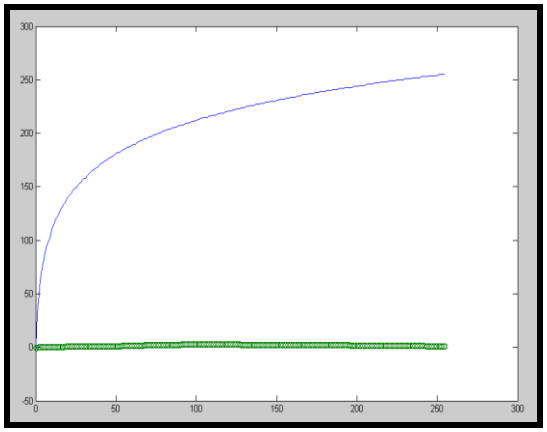

Fig. 9: *Simulation Output without Training.*

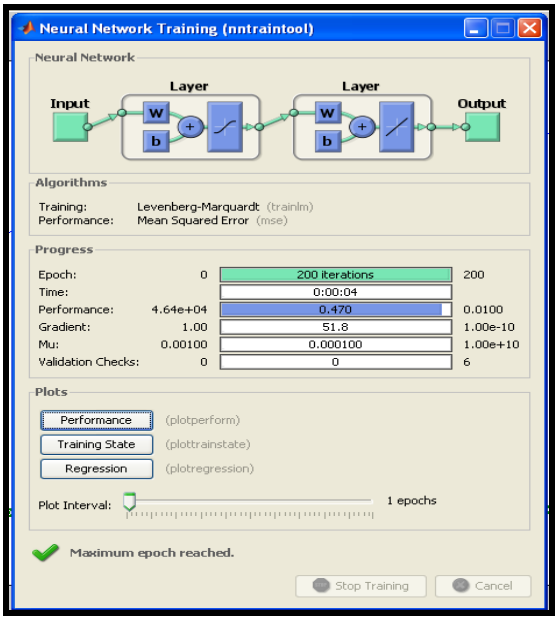

Fig. 10: *Neural Network Training.*

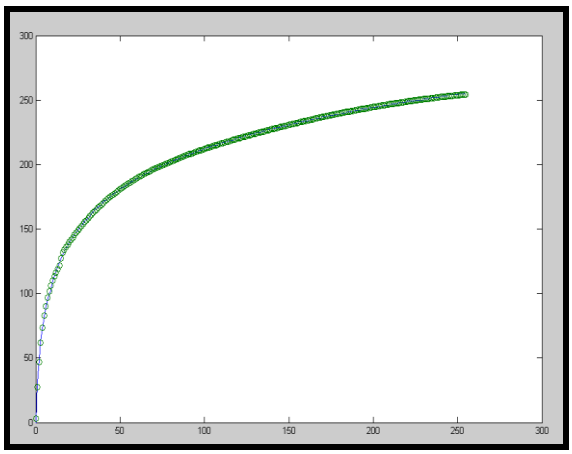

Fig. 11: *Simulation Output with 0:255 Input Range after Training.*

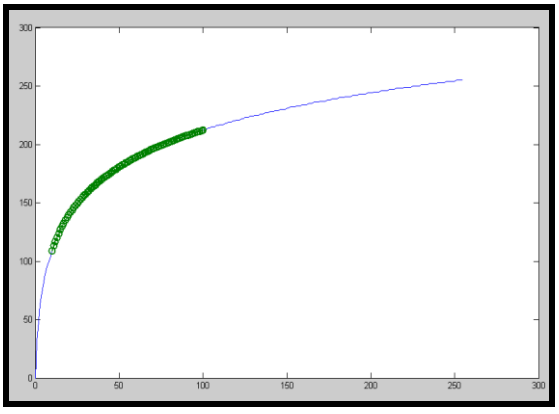

Fig. 12: *Simulation Output with 100:200 Input Range after Training.*

5. CONCLUSIONS

In this paper, we have presented the model for 8-bit Log ADC with neural network. From the result obtained by neural network, it can be concluded that the Log ADC, which we have practically implemented, shows the logarithmic response and the inverted graph is just the electronic property because with the same readings we get the same parabolic response as theoretical.

REFERENCES

1. A. Anand Kumar. *Fundamentals of Digital Circuits*. Prentice-Hall of India Private Ltd. Sixth printing, May 2004. Ch. 13.
2. http://www.allaboutcircuits.com/vol_4/chpt_13/7.html
3. Jongwoo Lee. *A Report on a Closed-Loop Deep Brain Stimulation Device with a Logarithmic Pipeline ADC.*
4. C. Bishop. *Neural Networks for Pattern Recognition*. Prentice Hall. 2nd Edn. 1998.
5. J. Zurada. *Introduction to Artificial Neural Systems*. Jaico Publishing House. 1998.
6. S. Russel and P. Norvig. *Artificial Intelligence: A Modern Apporach.* Pearson Education. 3rd Edn. 2010.
7. *Neural Network Toolbox Using MATLAB.* Matlab 7.01.
8. Jyh-Shing Roger Jang, Chuen Tsai Sun and Eiji Mizutani. *Neuro-Fuzzy and Soft computing – A Computational Approach to Learning and Machine intelligent.* PHI Learning Private Ltd. ISBN – 978-81-203-2243.
9. www.datasheetking.com/250--27C64+EPROM+programmer-d... - China
10. http://www.indiabix.com/electronics-circuits/log-amplifier/
11. www.analog.com/static/imported-files/tutorials/MT-026.pdf
12. www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/cs11/article1.html