

U-Tube Manometer Calibration using ANFIS

Hima Bindu S

Asst.Professor

Audisankara college of Engg
Nellore

Anitha Mary J

Asst.Professor

Karunya University
Coimbatore

Jegan R

Asst.Professor

Karunya University
Coimbatore

ABSTRACT

This paper deals with the implementation of the Adaptive neuro fuzzy inference system (ANFIS) on a Xilinx based Field Programmable Gate Array Spartan-3E. The implemented hardware is then used to efficiently calibrate the U-Tube manometer, in which the relation between the level of mercury and the Capacitance developed across the Copper Plates of the manometer is highly non-linear. This system shows that the ANFIS can be implemented in hardware and can be used to calibrate the non-linear systems effectively.

General Terms

Calibration, Artificial intelligence, Real time applications of artificial intelligence, ANFIS, Training algorithms for ANFIS, sigmoid function, Floating Point Arithmetic Unit

Keywords

ANFIS, Field Programmable Gate Array (FPGA), Non-linear Systems, Calibration, sigmoid function, Artificial Intelligence, ANFIS implementation on FPGA, Pipelined floating point arithmetic unit design

1. INTRODUCTION

ANFIS find their applications in solving the non-linear relations. Training the ANFIS is considered to be the one of the most challenging one. Training involves selection of an appropriate training algorithm, selection of the training patterns and floating point arithmetic for high efficiency.

Most of the ANFIS realizations till date are implemented in the software as in [1]. Contrary to this an efficient hardware implementation of the ANFIS in an FPGA is presented in this paper.

A non-linear system is a system in which the relation between the input and output are non linear. The major problem in the non-linear systems is that we can't estimate the output by traditional methods if we don't know the non-linear constant involved. One way to solve this type of problem is going for an ANFIS [2]. By implementing the ANFIS in the hardware, we can integrate this block into the non-linear system. This adds the flexibility of calibrating the system before its usage. As the hardware can be integrated into the system, we can calibrate it again if some of the dependencies of the system are changed. For example if a non-linear system has a dependency on temperature, the system can be calibrated one more time whenever we feel the temperature is changed.

In the U-tube manometer system copper plates are connected in such a way that the air/mercury will act as dielectric in between the copper plates. The capacitance across the copper plate varies whenever the level of mercury varies. The level of the mercury is used in calculation of pressure. The relation between the level of mercury and the capacitance developed is highly non-linear in this case. This makes it difficult for the system designer to calibrate it efficiently.

In this paper, the ANFIS based calibration is applied to the U-tube Manometer system, to demonstrate the application of ANFIS to estimate the output of non-linear systems.

2. ANFIS BASICS

Learning is the fundamental form of adaptation. ANFIS has the ability to learn. This is the primary reason for using the ANFISs where the solutions can't be obtained using traditional methods.

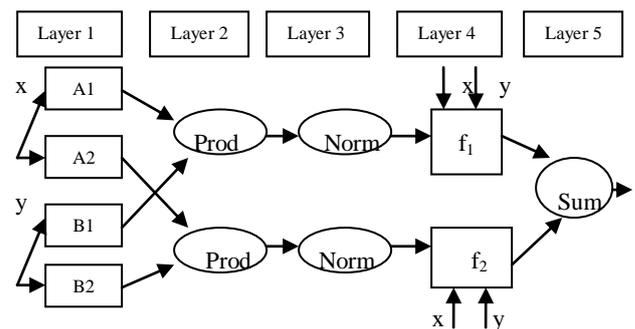


Fig 1: ANFIS model with 2 inputs x and y

ANFIS combines the advantages of both neural network and fuzzy logic which offers good results. Learning duration of ANFIS is very small as the output result depends directly on the firing strength of the fuzzy rule.

ANFIS is five layers architecture (See in Figure 1). ANFIS uses Sugeno Fuzzy model in which the rule set is given by

- Rule 1: If x is A₁ and y is B₁, then f₁ = p₁x + q₁y + r₁
- Rule 2: If x is A₂ and y is B₂, then f₂ = p₂x + q₂y + r₂

For a zero order sugeno model, we will consider p₁=q₁=p₂=q₂=0. Layer 1 output is the typical membership function with the parameters w₁, w₂ for A₁ and w₃, w₄ for A₂. The membership function can be taken as the difference of two sigmoid functions. Layer 2 output is the product of the outputs from the Layer 1. If we have only one input, Layer 2 output is same as layer 1 output. Layer 3 output is the normalization output by normalizing the outputs from layer 2, which gives the firing strength of the rule. This strength is then multiplied by the sugeno rule, which is the layer 4 output. The output level f_i of each rule is weighted by the firing strength w_i of the rule. The final output of the system is the weighted average of all rule outputs, given by

$$\text{Final Output} = \frac{\sum_{i=1}^n w_i f_i}{\sum_{i=1}^n w_i} \quad (1)$$

The Basic components are

- Membership function
- Feed forward Network
- Error correction.

The brief explanation of each of the component is as below:

2.1 Membership Function

Membership function is the basic component of an ANFIS. The input values to an ANFIS usually will be in the range of 0 to 1.

The Membership function used in almost all the ANFIS is the difference of two sigmoid functions. Let's consider x as the input to the Sigmoid function and $S(x)$ as the output then the sigmoid equation is given by

$$S(x) = 1/(1 + e^{-x}) \quad (2)$$

The sigmoid function has the derivative given by

$$S'(x) = S(x) * (1 - S(x)) \quad (3)$$

Then the membership function is given by

$$A(z) = S(w_1 * x) * S(w_2 * x) \quad (4)$$

Where w_1, w_2 are the arbitrary weights of membership function.

2.2 Feed Forward Network

In the Implementation of Feed forward network for ANFIS, Zero order sugeno model is used. Layer1 finds the product of the incoming signals from the input layer, layer3 normalizes the value from the output of layer2, layer4 approximated partial output based on firing strength of the rule and layer5 is the feed forward output of ANFIS which is summation of partial outputs.

2.3 Error Correction

If w_1, w_2, w_3, w_4 are the parameters of the ANFIS network (membership function), X_1, X_2 are the output of layer2, Y_1, Y_2 are the output of layer3, F_1, F_2 are the sugeno rules output Then for zero order sugeno model

$$F_1 = r_1 \quad (5)$$

$$F_2 = r_2 \quad (6)$$

When we want to train the network between two specific levels then we can set the sugeno rules output r_1, r_2 as our desired outputs, and then compute the weights w_1, w_2, w_3, w_4 by finding solutions to the following equations

$$X_1 = 0 \quad (7)$$

$$X_2 = 0 \quad (8)$$

Where X_1, X_2 are the outputs of membership functions. This is valid because for a given input if we make $X_1=0$, then the output will be exactly r_2 and when we make $X_2=0$, then the output will be exactly r_1 . The reason behind this is that the sum of Y_1, Y_2 is equal to one thereby when we make either X_1 or X_2 equal to zero then firing strength Y_1 or Y_2 will become 1 giving the desired output.

3. HARDWARE IMPLEMENTATION

The hardware implementation of ANFIS consists of two major modules, floating point arithmetic unit (FPAU) and the Step by step ANFIS. Since the hardware realization of the Sigmoid function over a range is hard to realize, the second order approximation function as described in [6] is used. The equations are again presented here for convenience. If we consider $S(x)$ as the output of the sigmoid function then

$$S(x) = 0 \text{ if } x < -1 \quad (9)$$

$$S(x) = 0.5 * (1 - x) * (1 - x) \text{ if } -1 < x <= 0 \quad (10)$$

$$S(x) = 1 - 0.5 * (1 - x) * (1 - x) \text{ if } 0 < x <= 1 \quad (11)$$

$$S(x) = 1 \text{ if } x > 1. \quad (12)$$

Equations (9)-(12) are very easy to realize in the hardware and the characteristics of this function are similar to the sigmoid function in the range of -1 to 1. The error margin is very low when we use these equations as discussed in [6].

An FPAU is designed and is used to evaluate all the equations in this paper. Since we are going for floating point arithmetic it adds up high degree of accuracy in calculations. There by the overall results of the system are more accurate.

3.1 Floating Point Arithmetic Unit

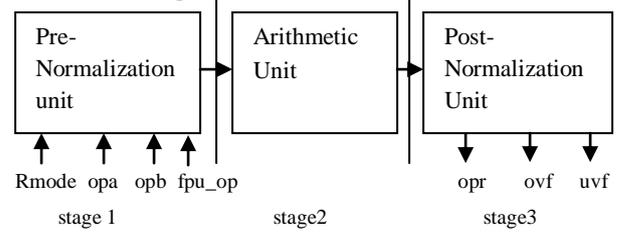


Fig 2: Three stage pipelined FPAU

A floating-point unit (a math coprocessor) is a part of a computer system specially designed to carry out operations on floating point numbers. Typical operations are addition, subtraction, multiplication, division. The IEEE-754 format is the most widely used floating point representation. The 32-bit IEEE-754 format for floating point is considered as the single precision floating point number representation. We implemented a single precision floating point arithmetic unit.

We designed a three stage pipelined architecture for the FPAU. Block diagram of the FPAU designed is as shown in Fig. 2. In Fig. 2 the inputs are 'rmode' is the rounding mode, opa and opb are the operands, fpu_op is the operation that needs to be carried on operands.

Pre-normalization units adjust the fractions (mantissa) and exponents. The Arithmetic Unit actually performs the addition, subtraction and multiplication based on the results from stage1.

A shared post normalization block, normalizes the fraction, and then rounds it. The final result is packed in to a valid single precision floating point format result. opr is the output result and ovf and uvf are signals generated to indicate overflow and underflow respectively. Overflow is signaled when the result can't be accumulated in a 32 bit format. Underflow is signaled when the result is too small to represent.

The operations addition, subtraction and multiplication are supported in the FPAU. The algorithms used are taken from [7] and [8]. The internal architecture of each of these stages is also taken from [7] and [8]. Apart from taking the concept from [7] and [8] the pipelining is designed with the above three stages for the single cycle execution of the FPAU unit.

For the sake of Normalization of the input a partial division algorithm which is the recursive division algorithm is also implemented as a separate unit. As the Normalized input values are in between 0 and 1 only this separate division unit is targeted in giving results in between 0 and 1 only in the IEEE-754 format for subsequent operations.

Table 1. Device utilization summary for FPAU

| S. No | Selected Device : 3s250etq144-4 | | |
|-------|---------------------------------|------------------|---|
| | Description | Occupied area | Percentage of occupied area to total area |
| 1 | Number of Slices | 624 out of 2448 | 26% |
| 2 | Number of Slice Flip Flops | 258 out of 4896 | 5% |
| 3 | Number of 4 input LUT's | 1067 out of 4896 | 22% |

| | | | |
|---|--------------------------|--------------|-----|
| 4 | Number of bonded IOBs | 7 out of 108 | 7% |
| 5 | Number of MULT18X 18SIOs | 3 out of 12 | 25% |
| 6 | Number of GCLKs | 1 out of 24 | 4% |

Table 1 shows the synthesis results of the FPAU implemented in a Xilinx Spartan-3E FPGA.

3.2 ANFIS

ANFIS and the error correction algorithm implementation on the hardware are carried out in several steps.

The ANFIS together with the error correction algorithm is implemented using the equations (1)- (12) each of which are evaluated using the FPAU unit designed.

An efficient mechanism is designed so that we can vary the input as well as the teacher output when we want to train for a particular pattern. This is implemented as a separate module with the instance of the FPAU called inside this. Duplication of the hardware is avoided by implementing both the feed forward network and error correction algorithm in the same module as both of them require the FPAU as a sub-module.

This ANFIS module takes in the normalized input value and the teacher output as the inputs in training mode and input value as the input in measurement/production mode. Output value is assigned only in measurement/production mode in which the training is already done.

Table 2. Device utilization summary for ANFIS

| S.No | Selected Device : 3s250etq144-4 | | |
|------|---------------------------------|------------------|---|
| | Description | Occupied area | Percentage of occupied area to total area |
| 1 | Number of Slices | 989 out of 2448 | 40% |
| 2 | Number of Slice Flip Flops | 548 out of 4896 | 11% |
| 3 | Number of 4 input LUTs | 1276 out of 4896 | 26% |
| 4 | Number of bonded IOBs | 4 out of 108 | 4% |
| 5 | Number of MULT18X18SIOs | 2 out of 12 | 17% |
| 6 | Number of GCLKs | 1 out of 24 | 4% |

Table 2 shows the device utilization summary for ANFIS using Spartan 3E.

3.3 U-Tube Manometer Calibration

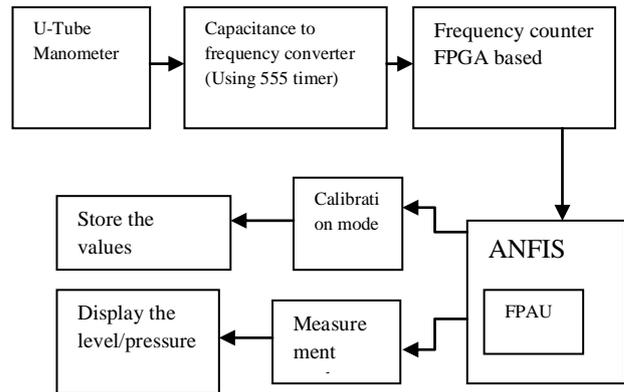


Fig 3: Block Diagram for U-tube Manometer Calibration

U-Tube manometer has a non-linear relation between the capacitance developed by changing the level of mercury and the level of mercury. Since the frequency produced by an astable multivibrator is inversely proportional to the capacitance across it, the capacitance can be converted into frequency using an astable multi-vibrator which also has a non-linear relationship with the level of mercury. The Astable multivibrator is designed using 555-timer. The capacitance developed across the copper plates of the manometer is converted into frequency because the FPGA can easily count the frequency generated.

As the frequency generated is in the range of Khz there is a need to normalize this frequency in the range of 0 to 1. Lets consider the Normalized frequency as F_{norm} , input frequency as F_{input} , maximum frequency in the training range as F_{max} , minimum frequency in the training range as F_{min} then the equation for normalizing this frequency is given by

$$F_{norm} = (F_{max} - F_{input}) / (F_{max} - F_{min}) \quad (13)$$

The partial floating point division block is used for this division and the result obtained is fed to the ANFIS for training or the feed forward case.

Fig. 3 shows the block diagram of the U-tube manometer calibration system.

The U-tube calibration/measurement module actually works in two modes, training mode and measurement mode. For this reason there is a control provided to switch to training mode or measurement mode.

In the training mode we have to feed the frequency to the FPGA kit corresponding to the training levels we have fixed at first. These frequencies are stored in an array for further processing. Once the number of levels we determined is done we will move on to measurement mode.



Fig 4: Setup for U-Tube Manometer Calibration

Table 3. Measurement mode table

| S.No | Frequency in Khz | Level in Cm |
|------|------------------|-------------|
| 1 | 27.68 | 0 |
| 2 | 26.70 | 1 |
| 3 | 26.02 | 2 |
| 4 | 25.37 | 3 |
| 5 | 24.85 | 4 |
| 6 | 24.29 | 5 |
| 7 | 23.76 | 6 |
| 8 | 23.24 | 7 |
| 9 | 22.75 | 8 |
| 10 | 22.28 | 9 |
| 11 | 21.9 | 10 |

Table 3 shows the teacher output values and the corresponding frequency levels which are used to train the network.

In both the training mode and measurement mode we will use a frequency counter to count the frequency. In training mode it's stored into an array where as if it's in measurement mode we will use this frequency to display its corresponding level. We use the concept given in [9] for the frequency counter.

In the measurement mode we will first find the upper limit and lower limit of the corresponding frequency with respect to

the Table 3. We will first train the network by giving the upper limit and lower limit input patterns to the ANFIS block and once it's trained, feed forward is done on the actual input frequency to obtain the result. Actually we are training the network between the two limits of the given frequency which is a piecewise approximation to obtain more accurate results rather than training the network for all the input patterns given in Table 3. Training for all the input patterns at once is prone to more error when compared with piece wise approximation because the non-linearity of the output will vary from point to point. Finally we will display the level of the mercury which is feed forward output on an LCD screen.

Table 4. Device Utilization Summary For U-tube Manometer Calibration

| S.No | Selected Device : 3s250etq144-4 | | |
|------|---------------------------------|------------------|---|
| | Description | Occupied area | Percentage of occupied area to total area |
| 1 | Number of Slices | 1989 out of 2448 | 81% |
| 2 | Number of Slice Flip Flops | 1544 out of 4896 | 31% |
| 3 | Number of 4 input LUTs | 3776 out of 4896 | 77% |
| 4 | Number of bonded IOBs | 17 out of 108 | 15% |
| 5 | Number of MULT18X18SIOs | 8 out of 12 | 66% |
| 6 | Number of GCLKs | 3 out of 24 | 12% |

Table 5. Calibration Mode Results Table

| S.No | Frequency in Khz | Level in Cm |
|------|------------------|-------------|
| 1 | 26.5 | 1.23 |
| 2 | 24.5 | 4.48 |
| 3 | 23.5 | 6.17 |
| 4 | 23 | 7.89 |
| 5 | 22.5 | 8.12 |

Table 4, Table 5 shows the synthesis result and measurement mode outputs of the U-tube manometer calibration module respectively.

Fig. 4 shows the setup picture used for the U-tube manometer calibration which consists of an astable multivibrator, U-tube manometer together with a Xilinx Spartan-3E FPGA.

4. CONCLUSION

The Proposed concept and the design are evaluated in the hardware using a Xilinx Spartan-3E FPGA. The proposed design of using ANFIS usage for calibration of non-linear systems opens the windows to go for the most accurate training algorithms to be implemented in the hardware.

By implementing the ANFIS in the hardware, we can integrate this block into the non-linear system. This adds the flexibility of calibrating the system before its usage. As the hardware can be integrated into the system, we can calibrate it again if some of the dependencies of the system are changed. For example if a non-linear system has a dependency on temperature, the system can be calibrated one more time whenever we feel the temperature is changed.

Future enhancements or studies can be carried out on the artificial intelligence algorithms for solving the non-predictive real time systems. We can improvise on the error correction algorithms used in Artificial Intelligence algorithms and can go for design or improvisation of the existing algorithms in terms of speed and accuracy.

5. ACKNOWLEDGMENTS

We wish to acknowledge staff, management and the professors of Karunya University for supporting us in providing the infrastructure and support for the completion of the project.

6. REFERENCES

- [1] Dhanalakshmi R, Palaniswami S " ANFIS based Neuro-Fuzzy Controller in LFC of Wind-Micro Hydro-Diesel Hybrid Power System",International Journal of computer applications, Vol. 42 , 2012.
- [2] Dr G S V P Raju, V Mary Sumalatha, K V Ramani, K V Lakshmi "Solving Uncertain Problems using ANFIS" , International Journal of computer applications, Vol. 1, No. 4, October 2010.
- [3] M. A. Denal, F. Palis and A. Zeghib " ANFIS Based Modeling and Control of Non-linear systems: A tutorial" IEEE International Conference on Systems, Man. and Cybernetics, pp. 3433-3438, 2004
- [4] Klir,G.J. and Yuan,B.(2000).Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice-Hall, India.
- [5] J.S.R. Jang " Adaptive Network- Based Fuzzy Inference. System" IEEE Trans. On Systems Man. and Cybernetics, Vol.23, No.3, pp 665-685, 1993.
- [6] M.T. Tommiska "Efficient implementation of sigmoid function for re programmable logic", IEEE Proceedings –computers and digital techniques, Volume 150, No 6, pages 403-411,Nov-2003.
- [7] Shao Jie, Ye Ning, Zhang Xiao-Yan "The Implementation of High-speed Floating-point DBF Based on FPGA",International Journal of Digital Content Technology and its Applications. Volume 5, Number 7, July 2011.
- [8] Subhash Kumar" Implementation of IEEE-754 Addition and Subtraction for Floating Point Arithmetic Logic Unit" International Transactions in Mathematical Sciences and Computer, Volume 3, Issue No. 1, 2010, pp. 131-140.
- [9] ken Chapman"Frequency Counter for Spartan-3E Starter Kit" Published in technical papers by xilinx on 7 th March 2006
- [10] MOHIT TYAGI "A Novel Hardware Efficient Reconfigurable 32-Bit Arithmetic Unit for Binary, BCD and Floating Point Operands", International Journal of Engineering Science and Technology (IJEST) , Vol. 3 No. 5 May 2011
- [11] Rabaey"Floating Point Arithmetic" in Digital Integrated Circuits, PHI publications
- [12] Ramakanth A. Gaykwad "Astable Multi Vibrator using 555 timer" in "Op-amps and Linear Integrated Circuits" book. PHI publications Rev 2006
- [13] Sawhney AK "U-Tube manometer operation and characteristics" in "Electrical and Electronics measurement and Instrumentation", Dhanpat Rai & Co, 2005