British Journal of Mathematics & Computer Science
14(4): 1-7, 2016, Article no.BJMCS.23014
ISSN: 2231-0851

SCIENCEDOMAIN *international*
www.sciencedomain.org

Valve Control System Using Fuzzy Logic

Mohamed Zaidan Qawaqzeh[1] and Yasir Khalil Ibrahim[2*]

[1]Al-Balqa' Applied University, Jordan.
[2]Jerash University, Jordan.

Authors' contributions

This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/BJMCS/2016/23014
Editor(s):
(1) Feyzi Basar, Department of Mathematics, Fatih University, Turkey.
Reviewers:
(1) Gopalakrishnan Jayapalan, UAE.
(2) Anonymous, Universiti Malaysia Sabah, Malaysia.
(3) Edita Lazarova, Slovak Academy of Sciences Kosice, Slovak Republic, Slovakia.
(4) Anonymous, Islamic University of Gaza, Palestine.
Complete Peer review History: http://sciencedomain.org/review-history/13266

Received: 10th November 2015
Accepted: 22nd January 2016
Published: 11th February 2016

Short Research Article

Abstract

The control of liquid level and flow between tanks is a basic problem in the process industries. Measuring the flow of liquids is a critical need in many industrial plants. Realizing lower development costs, better performance can be achieved using fuzzy logic which has better stability and fast response over other methods. The suggested solution aims to improve the control processes in a garden. The intended system can be used efficiently with high accuracy, low cost, a high quality.

Keywords: Fuzzy; controller; acquisition; PIC (Programmable Interrupt Controller); implementation; ultrasonic sensor.

1 Introduction

Fuzzy logic is a convenient way to map an input space to an output space. This is the starting point for everything else, and the great emphasis here is on the word "convenient". What do I mean by mapping input space to output space? Here are a few examples: You tell me how good your service was at a restaurant, and I'll tell you what the tip should be. You tell me how hot you want the water, and I'll adjust the faucet valve to

---

**Corresponding author: E-mail: dr.yasir.khalil@gmail.com;*

the right setting [1]. In this context, Fuzzy logic is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel PC or workstation-based data acquisition and control systems. Fuzzy controllers are very simple conceptually. They consist of an input stage, a processing stage, and an output stage. The input stage maps sensor or other inputs, such as switches, thumbwheels, and so on, to the appropriate membership functions and truth values. The processing stage invokes each appropriate rule and generates a result for each, then combines the results of the rules. Finally, the output stage converts the combined result back into a specific control output value. As shown in Fig. 1.

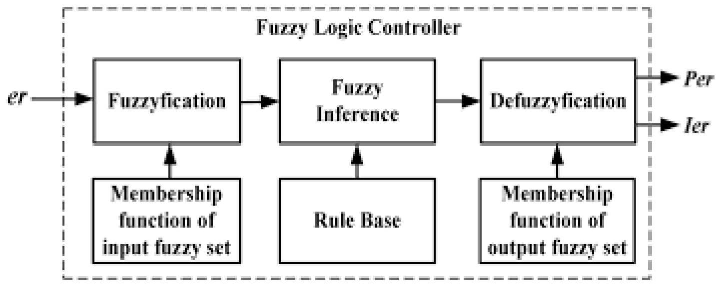


Fig. 1. Fuzzy logic controller block diagram

The processing stage is based on a collection of logic rules in the form of IF-THEN statements, where the IF part is called the "antecedent" and the THEN part is called the "consequent". Typical fuzzy control systems have dozens of rules.

Consider a rule for a thermostat: IF (temperature is "cold") THEN (heater is "high").

This rule uses the truth value of the "temperature" input, which is some truth value of "cold", to generate a result in the fuzzy set for the "heater" output, which is some value of "high". This result is used with the results of other rules to finally generate the crisp composite output. Obviously, the greater the truth value of "cold", the higher the truth value of "high", though this does not necessarily mean that the output itself will be set to "high", since this is only one rule among many [2].

2 Fuzzy Implementation

We want to control the degree of valve represented by the LCD depending on the temp and humid signals that came from the sensors and processed by the microcontroller. As shown in Fig. 2.

2.1 Fuzzy implementation steps

1) Choose suitable scale universe of discourse [3] of -L ≤ (temp, humid)≤ L.

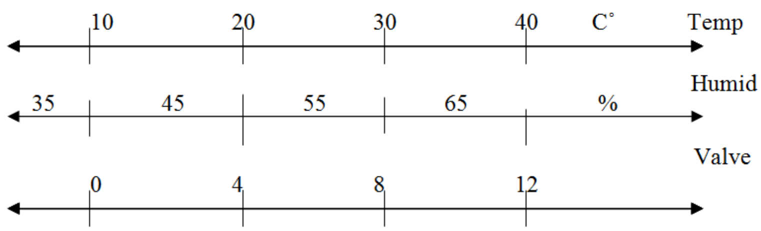
   Where L and –L represent the upper and lower limits of universe of discourse (U.O.D) of temp and humid.

Example:

   Temp range (10 – 40) C˚; U.O.D for temp
   Humid range (35% – 65%); U.O.D for humid
   Valve range (0 – 12); U.O.D for valve

2) Divide the U.O.D into equally spaced intervals which are called the non-fuzzy set intervals for temp, humid and valve.

Example:



The fuzzy set is define now. For our example let us take the following fuzzy set:

HT - High temp	HH - High humid	OV - Open valve
MT - Mid temp	MH - Mid humid	MV - Mid valve
LT - Low temp	LH - Low humid	CV - Close valve

Table 1. The fuzzy sets for temp, humid and valve

Fuzzy set		**Temp**	**10-20**	**20-30**	**30-40**
		Humid	**35-45**	**45-55**	**55-65**
		Valve	**0-4**	**4-8**	**8-12**
HT	HH	OV	0	0	0.8
MT	MH	MV	0.5	0.6	0.6
LT	LH	CV	1	1	0.1

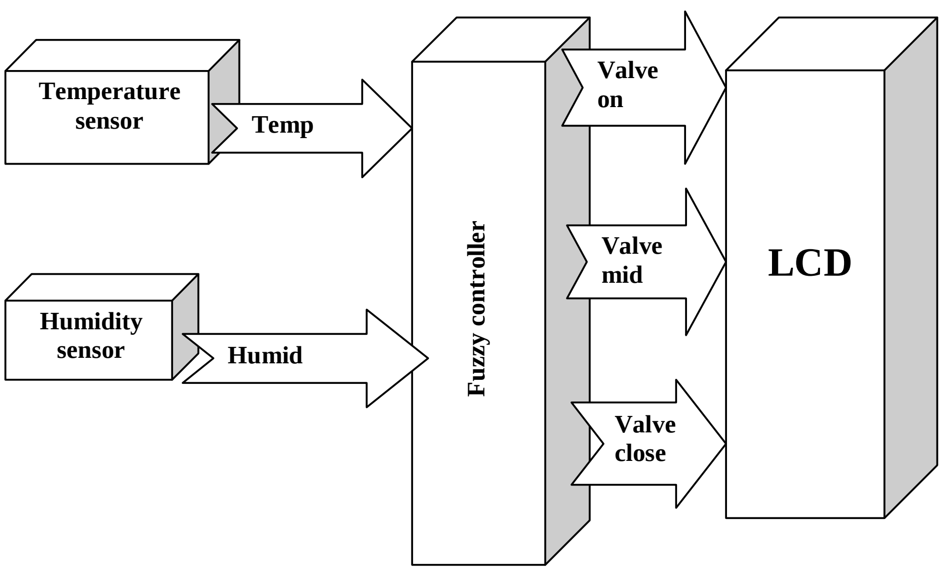


Fig. 2. Fuzzy controller block diagram

3) The decision rules are developed linguistically to do a particular control task and are implementing as a set of the form *IF HT and HH then MV*, see Table 2.

Table 2. Building rules

Temp / Humid	HT	MT	LT
HH	MV	CV	CV
MH	OV	OV	CV
LH	OV	OV	MV

4) The following steps can be used to calculate the scalar control action (the degree of danger).

   a) For temp and humid define the fuzzy subsets with their discrete membership function.
   b) Find the degree of freedom (D.O.F) from the fulfillment of the IF port for all rule by anding the membership of the both temp and humid. As shown in Table 3.

Example: IF temp = 35 C˚ and humid = 50%

(30 – 40)		(45 – 55)	
HT	0.8	HH	0
MT	0.6	MH	0.6
LT	0.1	LH	1

Table 3. Degree of freedom

Temp / Humid	HT 0.8	MT 0.6	LT 0.1
HH	MV	CV	CV
0	0	0	0 (→ D.O.F)
MH	OV	OV	CV
0.6	0.6	0.6	0.1
LH	OV	OV	MV
1	0.8	0.6	0.1

C) Calculate the control vector (UAj) for each rule by anding D.O.F with the control action subsets elements and as shown below.

<u>**UAj= D.O.Fj ∩ An**</u> , n=1,2…,3 number of interval, j =1,2,3,….,R number of rule.

Control vector of rule 1: 0 0 0
Control vector of rule 2: 0 0 0
Control vector of rule 3: 0 0 0
Control vector of rule 4: 0 0 0.6
Control vector of rule 5: 0 0 0.6
Control vector of rule 6: 0.1 0.1 0.1
Control vector of rule 7: 0 0 0.8
Control vector of rule 8: 0 0 0.6
Control vector of rule 9: 0.1 0.1 0.1

d) Compute the net control action [6] (UAnet) by oring the vectors UAj as follow:

R1 =	0	0	0
R2 =	0	0	0
R3 =	0	0	0
R4 =	0	0	0.6
R5 =	0	0	0.6
R6 =	0.1	0.1	0.1
R7 =	0	0	0.8
R8 =	0	0	0.6
R9 =	0.1	0.1	0.1
UAnet	0.1	0.1	0.8

OR

5) Calculate the scalar control action [5] (valve degree) using the center of gravity method in which the deterministic O/P has a vector value that divide the area under fuzzy set into two equal value.

$$Valve\ degree = \frac{sum(I * (weight)n) from\ 1\ to\ n}{sum(weight)n\ from\ 1\ to\ n}$$

Where, weight represents elements of the net control action vector and I represents the value of the interval n.

Valve of degree = ((2*0.1) + (6*0.1) + (10*0.8))/ (0.1+0.1+0.8)
Valve of degree =8.8

2.2 System design

Fig. 3 shows a simplified block diagram of the overall system. It is clear the PIC is connected to a MAX232 buffer which is connected to the PCs RS232 port. Note that the clock is specified as 20MHz this was not the original plan, as a slower clock speed would reduce power consumption (critical if using battery power supply) and less noise would have been generate; hence ADC readings would be more accurate, allow it is possible to put the PIC to sleep while taking the ADC reading, the ADC will send an interrupt waking the PIC once the acquisition is complete. This is not an option for this application because it takes a long time for the PICs oscillator to return to full speed, after a sleep operation; hence this means that the CRC display would not be refreshed fast enough to avoid flicker.

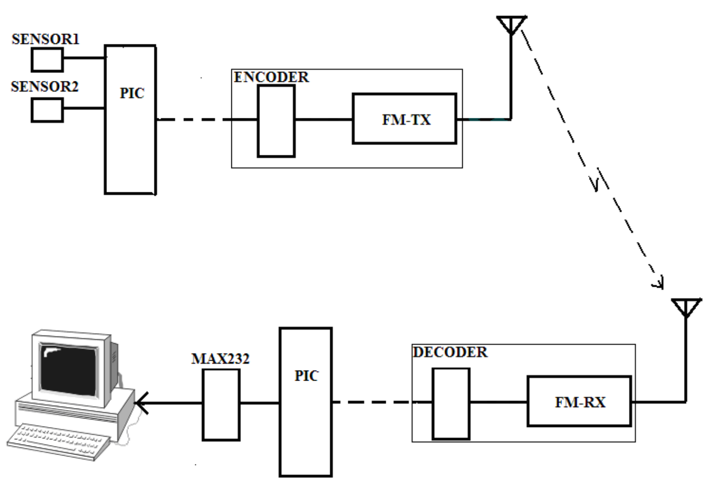


Fig. 3. Simplified block diagram of the system

The main reason for using a 20MHz clock was because the PIC program was developed in C, normally a C program will take 2-times longer to execute than if the program was written in assembly code. The chosen PIC is extremely powerful with a large program store (8K) and has no problem running C programs, making it ideally suited for the high-level programming language C. Microchip's 18 series (40MHz) are more powerful (design especially for C), these chips are much more expensive than the 16 series, hence were not a real option for this design [4].

Flow charts

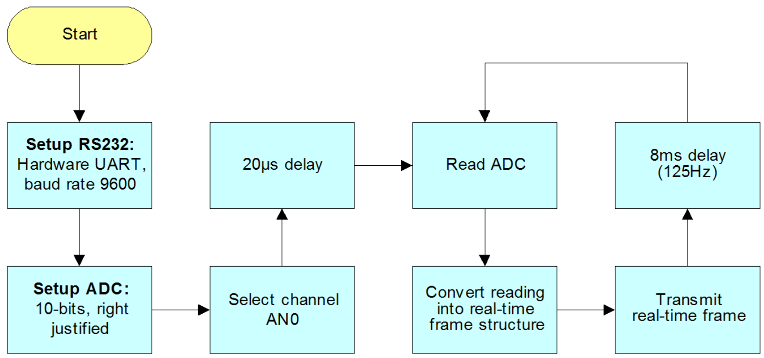


Fig. 4. Flow chart of main programming

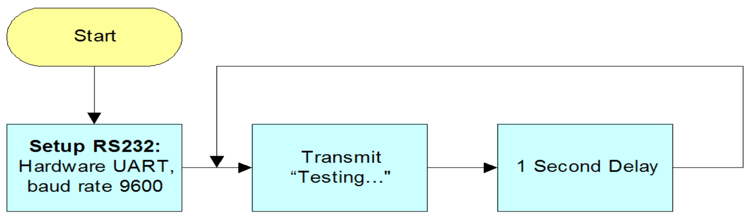


Fig. 5. Flow chart of RS232 communication programming

3 Results and Conclusions

The main results and conclusions we obtained from this work are summarized as follows:

- Our system provides a solution that helps the farmer to make a high speed and efficient irrigation control on the farm.
- This system can't be limited with just this two parameter (humidity and temperature) it depend on the parameter you like to measure and number of sensors used.
- Adjusting the sensitivity and the accuracy for the measuring circuits is important to ensure that the measured parameters represent its actual values.
- The use of microcontroller reduces the hardware components as it can used to do the function of more than one circuit.
- The microcontroller PIC16F877 was chosen; because of its low-cost and small size.
- The main advantage to use the serial port rather than parallel; there is no need for many wires as parallel transmission [6].

4 Future Work

After complete the main phase of our work, we suggest additional features that if added to this work will improve the quality and flexibility.

- We can regulate the values of some quantities such as temperature, and to send these values via SMS to other mobile.
- We can apply this system by measuring water level parameter in the tank, and using the ultrasonic sensor, in addition to valve.

Acknowledgement

The authors would like to thanks both Al Balqa' Applied University & Jerash University for supporting this research.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Ali MM. Neuro-fuzzy controller design. Ph.D. Dissertation, Control and Computer Engineering Department, University of Technology, Baghdad, Iraq; 1998.

[2] Curtis D. Johnson. Process control instrument technology, fifth edition. Prentice Hall International; 1997.

[3] Surachai Panich. Development of fuzzy controller for water level in stream boiler tank. JCS. 2010;6(11):1233-1236. Science Publications, ISSN 1549-3636.

[4] Dharamniwas and Others. Liquid level control by using fuzzy logic controller. International Journal of Advances in Engineering & Technology; 2012. ISN 2231-1963.

[5] Thae Thae and Others. Design 7 development of fuzzy logic controller for liquid flow control. IJECSE. 2012;3:208-218. ISN: 2277-1956.

[6] Elammari A. Design of heuristic fuzzy logic controller for liquid level control. Proceedings of Fifth International Conference on Intelligent Systems, Modelling and Simulation; 2014.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
http://sciencedomain.org/review-history/13266