International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-4, Issue-5)

Research Article

May 2015

Modeling, Design and Implement of Fuzzy Logic Controlleron FPGA Robotic Platform

Narendra M. Pathade, Prof. Sania M. Ansari
Department of ExTC & Pune University
Maharashtra, India

Abstract-

In this work we design, simulate and implement two fuzzy-logic based algorithms for mobile robots: one for obstacle avoidance, and another one with the combined objective of avoiding obstacles and as well as reaching a pre-defined target point in an unknown environment. The hardware used in this project is the National Instruments (NI)'s embedded robotic platform which houses the SBRIO (Single-board Reconfigurable Input-Output) that includes a powerful real-time controller, and a field programmable gate array (FPGA). For obstacle avoidance the robot has only one rotating ultrasonic sensor on the front side. The software is implemented using high-level LabVIEWmodules for embedded FPGA and real-time programming. Results show the key advantages of this new approach which is its accuracy, simplicity and quicker reaction to sudden changes especially when the robot is moving in an unstructured environment. This is due to the fact that the approach accounts for the size and shape of the robot and generates the speed of motion proportional to the distance from the obstacle.The hardware descriptive language used for the purpose of design of the Fuzzy Logic Controller is VHSIC Hardware Descriptive Language (VHDL) and is implemented on SPARTAN-3xc3s500e-4fg320FPGA board. This method of speed control of a dc motor may represent an ideal application for introducing the concepts of fuzzy logic. Here a sincere effort is made to show, how a fuzzy controller can be used to control the motor speed, which represents a very practical class of engineering problems.

Keywords -FLC, VHDL FPGA, DC motor, Hardware Implementation.

I. INTRODUCTION

The past few years have witnessed a rapid growth in the number and variety of application of fuzzy logic. The application ranges from consumer products such as cameras, washing machines, cars and in industry for medical instrumentation, underground trains and robots. Unlike the conventional controller FLC design is not based on the mathematical model of the plant or system. A FLC is an automatic controller that controls an object in accordance with desire behaviour. For a complex system whose mathematical model is very difficult to define or the transfer function of a plant is undefined, fuzzy logic controllers are very useful in that case [1,3]. The control action of FLC is defined in terms of simple human friendly "if – then rules". These set of rules are describe the system behaviour. These set of rules are called the knowledge base of fuzzy controller. We can easily change the rules accordance with our desire output. So the development time for a new controller can be significantly reduced as compared to conventional one [5].

The motivation behind the implementation of a FLC in VHDL was driven by the need for an inexpensive hardware implementation of a generic fuzzy controller for use in industrial and commercial applications [13]. We have taken a simple FLC for an armature control DC motor speed control. Error and change in error in speed has been used as two inputs to FLC. For both the inputs 5 triangular membership function has been selected and coded in VHDL. An algorithm has been developed in VHDL to fuzzifie the crisp digital values of speed error and rate of change of error. Sugeno type FLC structure has been used to obtain the controlled output. The controller algorithm developed synthesized, simulated and implemented on FPGA board. The FLC has been design using system generator approach.

A notable effort in this regard was made by NI with the introduction of its SBRIO board which combines deployable,embedded devices that feature a real-time processor, reconfigurable FPGA, and analogand digital input-output ports, all residing on a single-board programming in the NILabVIEW software [7, 8]. This software provides a powerful graphical development solution for designing, prototyping and deploying embedded systems. The NI-SBRIO based platform therefore provides a welcoming and important addition to the design and prototyping arsenal for embedded systems engineering. Moreover, the NI platform allows for its own hardware expansion via the addition of variety of analog and digital sensors, motion devices, and communication modules as well as for the enables the extension of its applicability area through designing and programming its own custom FPGA based module.

II. FUZZY LOGIC CONTROLLER DESCRIPTION

Fuzzy logic control is a control algorithm based on a linguistic control strategy, which is derived from expert knowledge into an automatic control strategy. Fuzzy logic control doesn't need any difficult mathematical calculation like the others control system. While the others control system use difficult mathematical calculation to provide a model of the controlled plant, it only uses simple mathematical calculation to simulate the expert knowledge. Although it doesn't need any difficult mathematical calculation, but it can give good performance in a control system. Thus, it can be one of the best available answers today for a broad class of challenging controls problems. A fuzzy logic control usually consists of the following as inFig. 1:

1) Fuzzification: This process converts or transforms the measured inputs called crisp values, into the fuzzy linguistic values used by the fuzzy reasoning mechanism.
2) Knowledge Base: A collection of the expert control rules (knowledge) needed to achieve the control goal.
3) Fuzzy interface engine: This process will perform fuzzy logic operations and result the control action according to the fuzzy inputs.
4) Defuzzification unit: This process converts the result of fuzzy reasoning mechanism into the required crisp value.

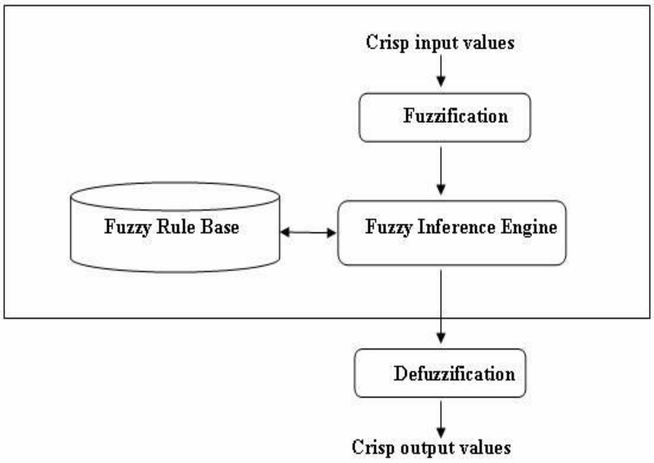


Fig.1. Fuzzy logic controller

III. ALGORITHM FOR OBSTACLE AVOIDANCE

In the first paper a fuzzy logic based approach for robot obstacle avoidance has been developed and implemented on NI's SBRIO 2-wheel differential drive robotic platform. It uses the ultrasonic sensor mounted on a motor to see how far the obstacles are in front of the robot up to a maximum distance of 1m. The ultrasonic sensor can turn left and right up to 65 degrees on each side (hence sweeping through a 130-degrees frontal area) in order to detect the obstacles in front of the robot. The robot is a 2-wheel differential-drive mobile robot. In order to design this new approach that should provide smoother motion and instant reaction to sudden changes, we started by analyzing the demo program (called 'roaming') that captures 65 values of obstacle distances with a 2 degrees step. It then finds the largest gap from these values and then let the robot move in that direction by giving command to its left andright motors up to a maximum angular speed of ±16 rad/sec. In the literature, this approach is called Vector field histogram (VFH) [2, 10]. It is observed that this program is not very accurate and sometimes would ignore the obstacles that are close and hit these in order to reach the largest gap. Therefore a new fuzzy logic controller is designed, simulated in LabVIEWand implemented on SBRIO robot, and its overall performance is analyzed.

A. Design of Fuzzy controller

The algorithm is designed to use 3 distance linguistic values as inputs, corresponding to obstacle distance at -65degrees (Left), 0 degrees (Middle), and +65 degrees (Right) angle.

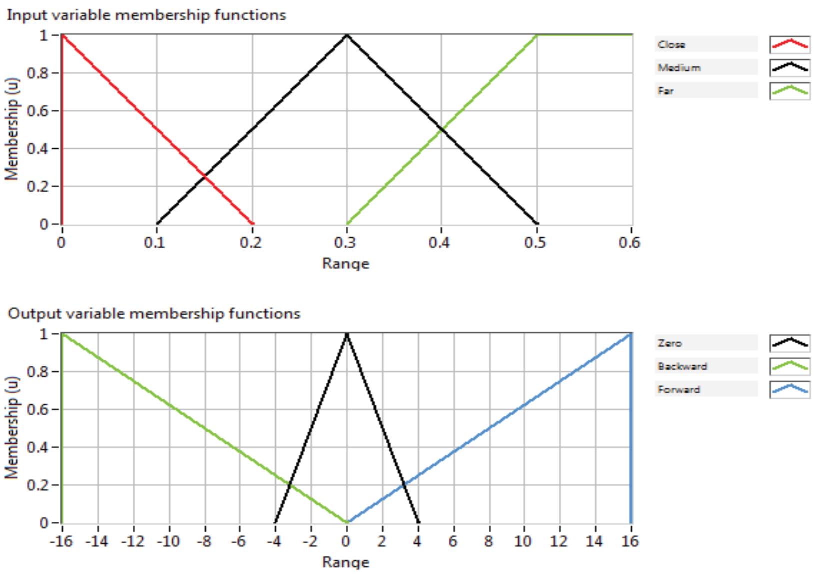


Fig 2. Fuzzy membership functions for the input/outputs

For implementation purposes the ultrasonic sensor is installed on the front side which rotates continuously and senses the distances from obstacles on the Left, Middle and the Right sides. The two outputs of the fuzzy controller are the speed of the Left and Right wheels after fuzzification of the 3 input variables, application of the fuzzy rules, and defuzzification. These inputs and outputs are fuzzified using the membership functions given in figure 2. The radius of the robot wheels are 5 cm.

The input distances are classified either as **Close** (membership value *u* decreases from 0 to 0.2 meter), **Medium**(from 0.1 to 0.5 m) and **Far** (*u* increases for the distance from 0.3 to 0.5 m, and then remain constant after that). Each output has three membership functions: **Backwards** (represents speedfrom -16 to 0 rad/sec, or -80 cm/sec to 0), **Zero** (speed from -4 to +4 rad/sec, or -16 to +16 cm/sec), and **Forward** (from 0 to +16 rad/sec, or 0 to 80 cm/sec).
Following are 8 of the rules, out of total 27 rules used forthe development of fuzzy controller.
1. IF 'Left' IS 'Far' AND 'Middle' IS 'Far' AND 'Right' IS 'Far' THEN 'Right' IS 'Forward' ALSO 'Left' IS 'Forward'
2. IF 'Left' IS 'Far' AND 'Middle' IS 'Far' AND 'Right' IS 'Medium' THEN 'Right' IS 'Forward' ALSO 'Left' IS 'Zero'
3. IF 'Left' IS 'Far' AND 'Middle' IS 'Far' AND 'Right' IS 'Close' THEN 'Right' IS 'Forward' ALSO 'Left' IS 'Backward'
4. IF 'Left' IS 'Far' AND 'Middle' IS 'Medium' AND 'Right' IS 'Far' THEN 'Right' IS 'Zero' ALSO 'Left' IS 'Zero'
5. IF 'Left' IS 'Far' AND 'Middle' IS 'Medium' AND 'Right' IS 'Medium' THEN 'Right' IS 'Forward' ALSO 'Left' IS 'Zero'
6. IF 'Left' IS 'Far' AND 'Middle' IS 'Medium' AND 'Right' IS 'Close' THEN 'Right' IS 'Forward' ALSO 'Left' IS 'Backward'
7. IF 'Left' IS 'Far' AND 'Middle' IS 'Close' AND 'Right' IS 'Far' THEN 'Right' IS 'Backward' ALSO 'Left' IS 'Backward'
8. IF 'Left' IS 'Far' AND 'Middle' IS 'Close' AND 'Right' IS 'Medium' THEN 'Right' IS 'Forward' ALSO 'Left' IS 'Zero'

B. Simulation in LabVIEW
The fuzzy logic controller with 27rules is implemented using the Fuzzy System Designer (FSD),and PID and Fuzzy Control toolkit of LabVIEW [11], bydefining in the toolkit the membership functions for theinputs/outputs, writing the rules and defuzzification procedure,where the defuzzification uses an approach called COA (Centerof Area) [12]. Finally this fuzzy controller is called from asimple LabVIEW VI. Above rule illustrates 8 different possibilities(out of 27) of obstacles sensed on the left, middle and rightside. In each case, each of the 27 rules has different degrees offulfillment (DOF) depending on the membership functiondefined in figure 1 (top picture) and the input distances involved. Figure 3 (top picture) shows the LabVIEW program written in order to implement this fuzzy controller. This program receives three distances on the Left, Middle and the Right side, and generates velocities for the two wheels. The simulation results for the input values read (Left, Middle and Right sensor), and the output values of speed generated for the Left and Right wheel for cases 5 and 7. It is clear that since for case-5 the obstacle is farthest on the left side, the right wheel should turn faster compared to the left wheel in order to move the robot to the left side. Similarly, for case 7 the obstacle is close on the left side, so the output command moves the roboton the right side.

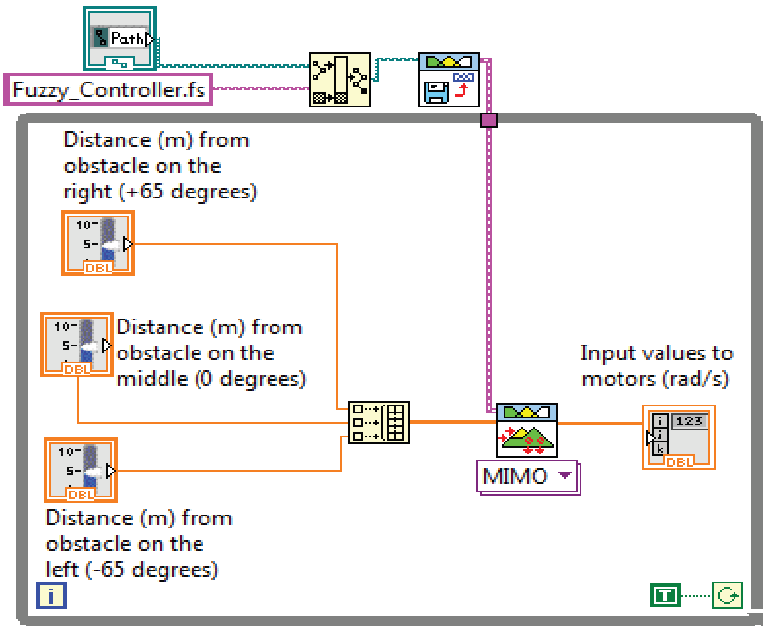


Fig 3.LabVIEW simulation code for fuzzy controller

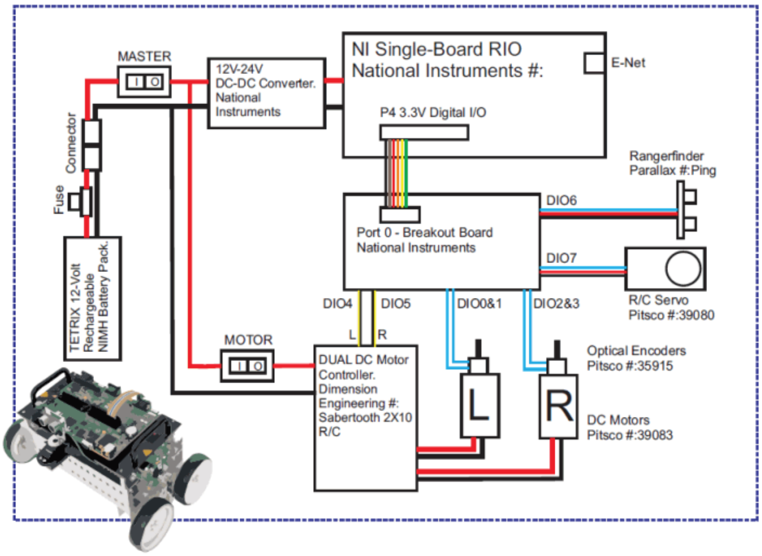


Fig 4. SBRIO Robot's physical structure

III. EXPERIMENTS AND RESULTS

Simulations have been done in Xilinx ISE.Fig.5 and Fig.6 shows that Test RTL view of FLC using VHDL and design summary respectively. In this figure the outputs have calculated according to input i.e. Error and Change in Error

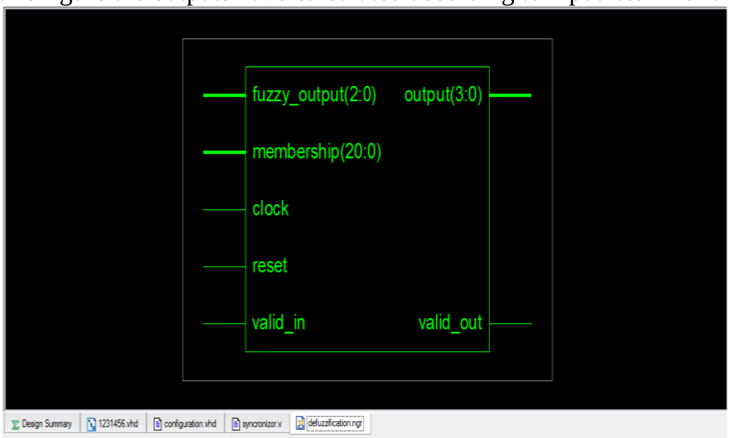
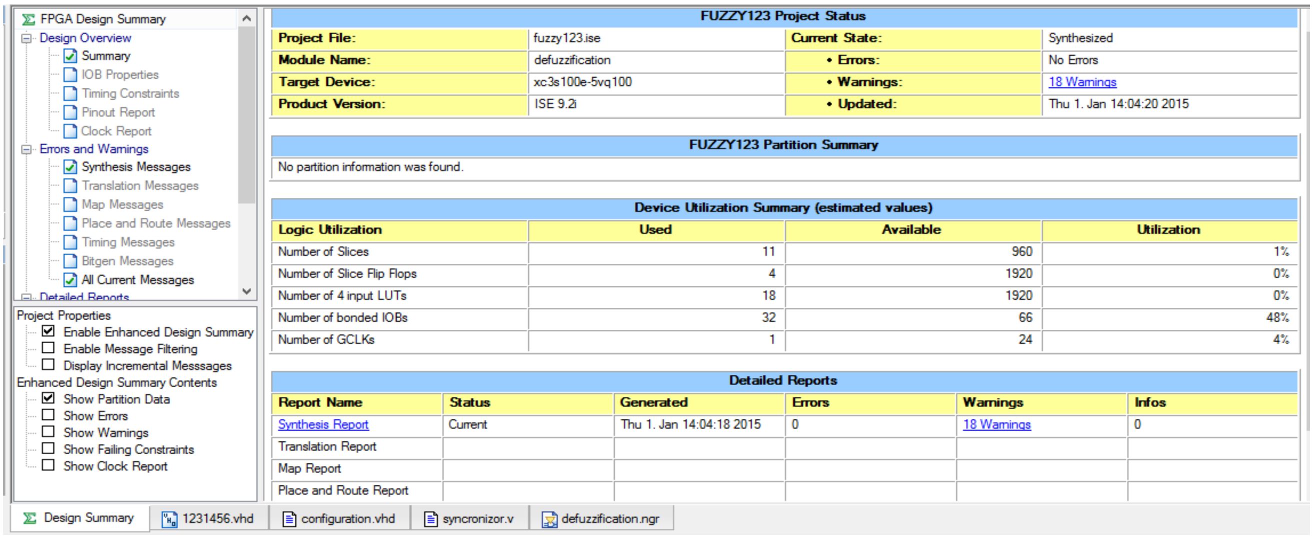


Fig.5. RTL view of FLC using VHDL



FUZZY123 Project Status			
Project File:	fuzzy123.ise	Current State:	Synthesized
Module Name:	defuzzification	• Errors:	No Errors
Target Device:	xc3s100e-5vq100	• Warnings:	18 Warnings
Product Version:	ISE 9.2i	• Updated:	Thu 1. Jan 14:04:20 2015

FUZZY123 Partition Summary
No partition information was found.

Logic Utilization	Used	Available	Utilization
Number of Slices	11	960	1%
Number of Slice Flip Flops	4	1920	0%
Number of 4 input LUTs	18	1920	0%
Number of bonded IOBs	32	66	48%
Number of GCLKs	1	24	4%

Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Thu 1. Jan 14:04:18 2015	0	18 Warnings	0
Translation Report					
Map Report					
Place and Route Report					

Fig 6.Design summery

We have taken a same 2nd order system for Dc motor to show the step response performance of FLC. Black box represents the FLC using VHDL code. The two input of FLC i.e. Error and Change in Error are going through the Gateway In to the system generator Black Box, the two outputs are coming through the Black Box. The control output of FLC is finally obtained by dividing one output by another.

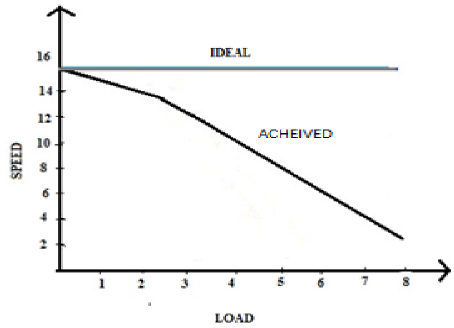


Fig. 7. Speed-Load Characteristics of the DC Motor obtained by implemented FLC Approach

IV. CONCLUSION

We develop two fuzzy-logic based algorithms for mobile robots: one for obstacle avoidance, and another one with the combined objective of avoiding obstacles and as well as reaching a pre-defined target point in an unknown environment. We also implement the first algorithm on NISBRIO robotic platform. Other testing is performed in LabVIEW simulation environment. The approach accounts for the size and shape of the robot and generates the speed of motion proportional to the distance from the obstacle in real-time. Overall, the approach is found to be accurate, simple to implement and results in quick reaction to sudden changes of obstacles especially when the robot is moving in an unstructured environment. The approach can be improved adding more sensors especially on the back side of the robot. Also, it is planned to use neural network to learn the environment while performing the mapping and obstacle avoidance operations.

REFERENCES

[1] S. Assilian and E.H. Mamdani,―An Experiment in Linguistic Synthesis with Fuzzy Logic Controller‖, Int. Journal on Man machine studies. Vol. 7,pp. 1-13,1975.
[2] R. Palm, ―Scaling of Fuzzy Controller using the cross correlation‖, IEEE Trans. Fuzzy Syst., Vol. 3, pp. 116-123, Feb. 1995
[3] H. X. Li and H. B. Gatland, ―Conventional Fuzzy Control and its enhancement‖, IEEE Trans. Syst., Man, Cyber., Vol. 26, pp. 791-797, 1996.
[4] Y. F. Li and C. C. Lau, ―Development of Fuzzy Algorithm for Servo System‖, IEEE Int. conference on Robotics and Automation, April 24-29, 1998.
[5] Andrew Kusiak,‖Fuzzy Logic‖, The University of Iowa, Iowa City 2004.
[6] Daijin Kim, ―An Implementation of Fuzzy Logic Controller on the Reconfigurable FPGA System‖, IEEE Transaction on Industrial Electronics, Vol. 47, No. 3, pp. 703-715, June 2000.
[7] Sameep and Kuldip S. Rattan, ―Implementation of a Fuzzy Controller on an FPGA using VHDL, 22nd International Conference (NAFIPS), pp. 110-115, March 2003.
[8] Philip T. Voung, Asad M. Madni and Jim B. Vuong, ―VHDL Implementation for a Fuzzy Logic Controller‖, World Automation Congress (WAC), July 24-26, 2006.
[9] Clive Max field, Design Warrior Guide to FPGAs, Elsevier publications 2004.
[10] Dr. Peter R. Wilson, Design recipes for FPGAs, Embedded technology series, Elsevier publication 2007.
[11] Moe Shahdad,―An Overview of VHDL Language and Technology‖. 23rd Conference on Design Automation, pp.320-326, 1986.
[12] YodyiumTipsuwan and Mo-Yuen Chow, ―Fuzzy logic Microcontroller Implementation For DC Motor Speed control‖, IEEE, pp. 1271-1276, 1999.
[13] DaviNunesOliveria, Gustavo Alves de Lima Henn and Otacilio da MotaAlmeida,‖Design and Implementation of a Mamdani Fuzzy Inference System on an FPGA using VHDL‖ Annual Meeting of the North American Fuzzy Information Processing Society, 2010.
[14] Marek J. Patyra, Janos L. grantner and Kirby Koster,‖Digital Fuzzy Logic Controller : Design and Implementation‖, IEEE Transaction on Fuzzy System, Vol. 4, No. 4, pp. 439-459, November 1996.
[15] ValentinaSalapura and Volker Hamann,‖ Implementing Fuzzy Control Systems Using VHDL and Statcharts‖, Design Automation Conference with EURO-VHDL, pp. 53-58,1996.

