Realization of Reconfigurable FLC on ADSP-BF537 Processor

P. Maji[†], S. K. Patra[†], K. K. Mahapatra[†], J. Govindarajan[§] and J. J. Patel[§]
[†]Dept. of Electronics and Communication Engineering
National Institute of Technology, Rourkela
Rourkela, Odisha, India
[§]Board of Research in Fusion Science and Technology
Institute of Plasma Research, Ghandhinagar
Gujarat, India.
Email: [†]{pallab, skpatra, kkm}@nitrkl.ac.in, [§]{govindarajan, jjp}@ipr.res.in

***Abstract*— Fuzzy Logic Controllers (FLC) provides excellent control to non-linear systems owing to its ability to represent linguistic and heuristic knowledge in mathematical formulation. However, this delineation comes at an expense of large no. of computations. This paper discusses Fired Rules Hyper Cube (FRHC) based rule reduction technique and implements it on Analog Devices ADSP-BF537 processor. Performance of the designed system is compared and analysed with Fuzzy Logic Toolbox of Matlab which expended as benchmark for testing using a simple control of DC motor.*)**

Keywords- ADSP-BF537, Fired Rules Hyper Cube, Matlab/Simulink Real-Time Workshop, Visual DSP++.

I. INTRODUCTION

Fuzzy logic inference systems provide a platform for programming qualitative and/or linguistic nature of control and decision making which are based on uncertainty and dubiety. However programming rules in a FLC is exponentially proportional to no. of inputs and membership functions of each input. If an FLC has N no. of inputs and each input has x no. of membership functions, then total no. of rules for that systems will be x^N. Hereby most important issue that needs to be addressed before designing any real-time FLC system is to avoid the standard rule base search that involves loops in the algorithm. FRHC was introduced by I. G. Kalaykov [1] provides an easy method for rule base reduction which reduces computations drastically.

In literature application specific [2, 3, 4] and FPGA based FLCs are dominant [5]. However the proposed design deals with file handling which is easier to implement in a DSP to implement configurability on run-time; such a system in FPGA is very complex and developing a synthesizable code for such complex algorithms is arduous. Designing of FLCs in DSP hereby provides characteristics of tuning and configurability in real-time.

Fig. 1 shows the block diagram of proposed system and the incorporation of text files as source of information for the system. Rulebase.txt provides knowledge base and Setup.txt provides the details of the fuzzy input and output membership functions. Data from these text files are transmitted serially to the ADSP-BF537 hardware platform and stored in memory. Run-time configurability is achieved by polling based interrupt service routine (ISR). Text files are open source requiring no third-party libraries to access them. Thus text format data were chosen in this implementation to keep complexity of the algorithm to minimum.

This paper is organized as next section deals with basics of Fuzzy Logic systems and Analog Devices ADSP-BF537 EZ-Lite kit which has been used as the hardware platform. Section III describes FRHC and section IV limns design specification, architecture and implementation. Section V discusses the performance analysis and the paper is concluded in section VI.

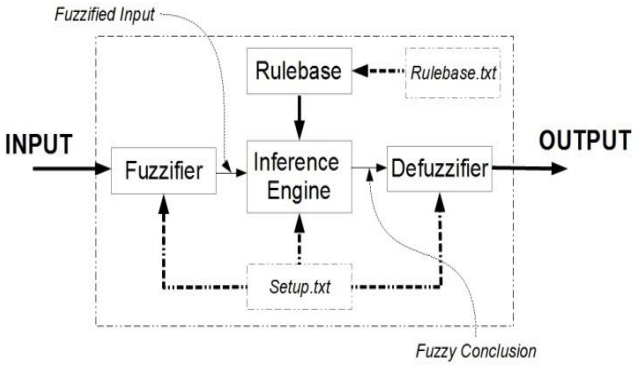

Fig. 1 Block diagram of proposed system

II. BACKGROUND MATERIAL

A. Fuzzy Logic Controller:

Fuzzy Logic Controller (FLC) does not associate with the mathematical model of any system, be it linear or non-linear. This feature is widely used for plants for which the mathematical modelling is catchy. It also provides the platform for programming linguistic knowledge in digital domain. These characteristic make FLC a very good controller for non-linear systems. FLCs are governed by a set of "if-then" rules, often termed as Rule Base, which governs the inference engine to bring forth fuzzy output in response to one or a set of inputs. The inputs are in generally real world analog signals called as Crisp input and are converted to fuzzy variables in Fuzzifier. Since most of the development platforms are digital, there is always an ADC premising the Fuzzifier. These fuzzified inputs proceed to the inference engine where controller response is generated using the rulebase as fuzzy output. There are various fuzzy based operators viz. min-max, product-max etc mentioned in the literature on which inference engine operates. These responses are then defuzzified to digital signals in Defuzzifier [6] if the controller platform is digital and if the desired output has to be in analog domain, a DAC succeeds the process. The rulebase can be derived from expert heuristic information or from numerical data. Performance of a FLC largely depends on the quality of its rulebase. Thus runtime configurability is very important for these systems. This paper explores a way of achieving this configurability in runtime serially using ISR based polling.

B. ADSP-BF537 [6]:

In literature, plenty of DSP based FLCs are present *[7, 4, 8]*. This work has been implemented on an Analog Devices' Blackfin fixed point 600MHz DSP with 132 KB of on-chip memory. The system peripherals include an IEEE-compliant 802.3 10/100 Ethernet MAC (ADSP-BF536 and ADSP-BF537 only), a CAN 2.0B controller, a TWI controller, two UART ports, an SPI port, two serial ports (SPORTs), nine general-purpose 32-bit timers (eight with PWM capability), a real-time clock, a watchdog timer, and a parallel peripheral interface (PPI) [ADSP Datasheet]. The hardware platform used is ADSP-BF537 Ez-Kit Lite which is programmed using C/C++ language and a cross compiler, VisualDSP++. Development of this system has been done using VisualDSP++ ver. 5.

III. FIRED RULES HYPER CUBE

Fired Rules Hyper Cube (FRHC) is characterized by extremely simple way of the fuzzy inference in a layered parallel architecture. The processing time slightly depends on the number of inputs of the fuzzy system and does not depend on the number of rules and fuzzy partitioning of all variables [FRHC, Kalaykov]. FRHC makes an assumption that the input membership functions are defined such that not more than two of them overlap at any given point of the input space.

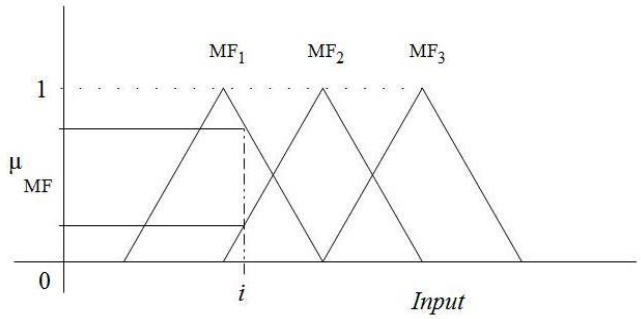

Fig. 2 FRHC Assumption

Fig. 2 has three membership functions namely MF1, MF2 andMF3. The output of Fuzzifier will be fuzzy set of three fractional values in between 0 and 1 for each input corresponding to distinct membership function. However, for any value of input i, there can be at maximum two non-zero values or conversely at minimum one non-zero value. This of course depends on the no. of overlapping membership functions which in FRHC is assumed to be two. At no value of the input, within the range, it is possible to obtain a nonzero fuzzy set. This property has been exploited in the design of the proposed FLC to enhance the fuzzy logic inference per second (FLIPS).

IV. HARDWARE IMPLEMENTATION:

A run-time configurable FLC is coded in C language and emulated in an ADSP-BF537 Blackfin Processor with help of Analog Devices VisualDSP++ software environment. Offline data are fed into the program through text files as shown Fig. 1 serially and the results are recorded for performance analysis. The algorithm is implemented in C language and compiled to generate an output file which can be dumped on to the Blackfin processor. The code size is 109KB which fits in perfectly in the stipulated internal memory of 132 KB resulting in no external memory to be employed.

The setup parameters are as follows:

No. of Input	:	1 (Speed)
No. of Output	:	1 (Voltage)
Input Membership Functions	:	3
Output Membership Functions	:	3
Shape of Membership Function	:	Triangular
Implication Method	:	Min
Aggregation Method	:	Max
Defuzzification Method	:	Weighted Average

These parameters were fed to the controller at startup through a text file named Setup.txt. Similarly the rulebase was fed to the controller through a text file named Rulebase.txt. After this the controller is ready to operate with specified set of inputs.

Standard defuzzification methods include.

- BADD (basic defuzzification distributions)
- COA (centroid of area)
- COG (bisector of area)
- FOM (first of maximum)
- LOM (last of maximum)
- MOM (middle of maximum)
- WFM (weighted fuzzy mean)

As mentioned earlier in the system design configuration, the proposed system was developed with a ***weighted average*** defuzzification method since this method is most widely used as defuzzification for control applications and produces meaningful control in most cases. Weighted average is calculated by,

$$Y^* = \sum_{i=1}^{N} \mu^i w_i$$

Where Y^* is the defuzzified output, μ^i is the membership value of output of each rule, w_i is the weight associated with each membership function and N is the total no. of output membership function. This method of defuzzification is very uncomplicated and easy to implement. A relatively fast inference time is observed and with fair amount of accuracy.

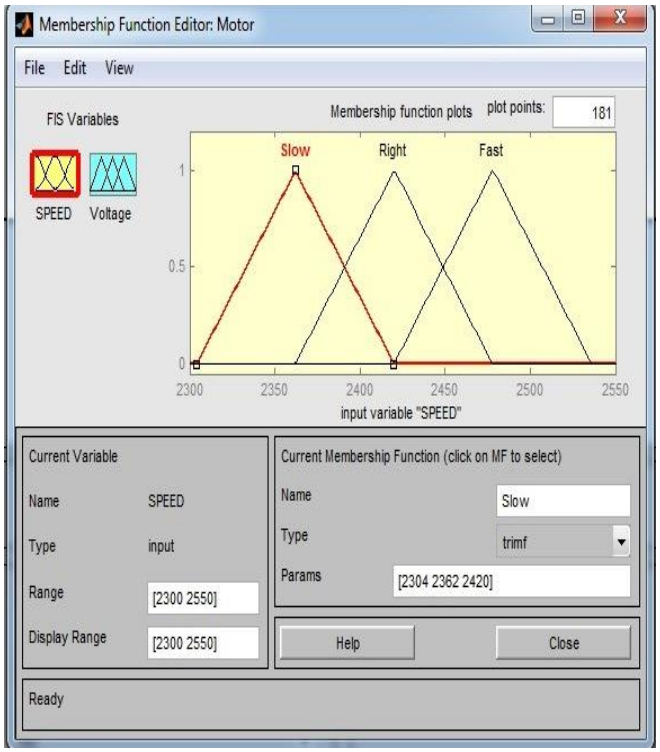

Fig. 3 Input Membership Function (Speed)

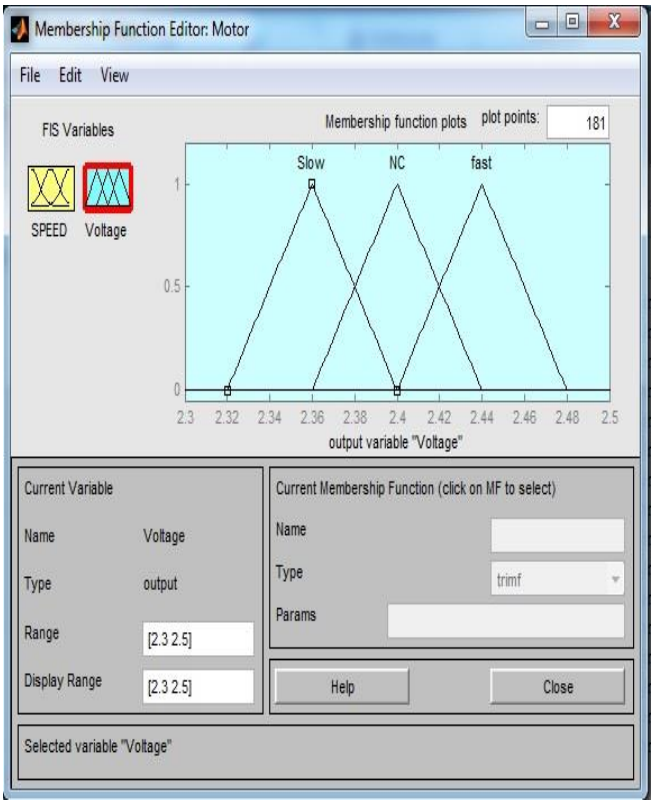

Fig. 4 Output Membership Function (Voltage)

TABLE I. COMPARISION RESULTS

Sl. No.	Speed (in RPM)	Blackfin Output (in volts)	Matlab Output (in Volts)
1.	2437.4	2.3880	2.3866
2.	2330	2.4620	2.4400
3.	2510.2	2.3382	2.3600
4.	2392.8	2.4188	2.4190
5.	2450	2.3802	2.3794
6.	2315	2.4620	2.4400
7.	2350	2.4620	2.4400
8.	2400	2.4142	2.4149
9.	2360	2.4620	2.4400
10.	2420	2.4000	2.4000

V. PERFORMANCE ANALYSIS

To test the developed system, a one-input one-output system of DC motor control is assumed. A FLC to control DC motor is developed in Matlab. Same FLC parameters are

fed to the hardware based FLC and the results are tabulated in Table I.

Fuzzy Inference Systems in Matlab is used to define the FLC which is used as a standard for testing the Blackfin based FLC. Fig. 3 and Fig. 4 shows the input and output membership functions for DC motor control where Speed (in RPM) is input and voltage (in V) is output. Fig. 5 shows the FLC setup parameters.

VI. Conclusion

A general purpose FLC based on ADS-BF537 is designed and analysed in comparison to Matlab FLC for performance and accuracy yielding the result to be satisfactory. The system exhibits a maximum of 0.5% of error due to truncation and sampling data bits. However majority of the error occurred due to weighted average defuzzification. Run-time configurability of the design was achieved using the file handling capability of Blackfin DSP. Text files were used to program the hardware with essential information about setup parameters and the rulebase. This system thus proves to be a runtime configurable general purpose FLC which can be programmed with expert heuristic knowledge and can be deployed in any control application, linear or non-linear. However this system lacks the ability of tenability of rulebase and operating in standalone mode. Implementation of such system will yield a great.

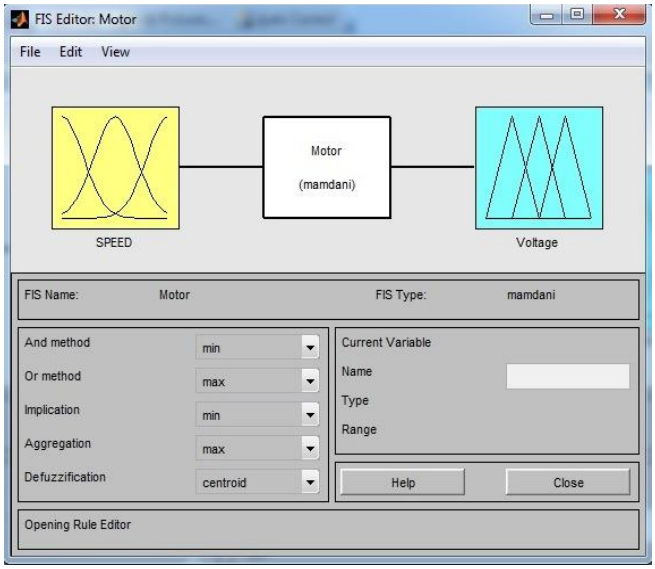

Fig. 5 Matlab FIS Structure

Acknowledgment

This work was supported by Board of Research for Fusion Science and Technology (BRFST), Gandhinagar, India. The authors would like to thank the people from IPR, Gandhinagar and NIT Rourkela, who have been indirectly involved in realizing this work. The authors are also thankful to anonymous reviewers for their valuable comments.

References

[1] I. Kalaykov, "New speed limits of the fuzzy controller hardware," in *Circuits and Systems, 1999. 42nd Midwest Symposium on*, 1999.

[2] M. K. El-Sherbiny, G. El-Saady, E. A. Ibrahim And A. M. Sharaf, "Efficient Incremental Fuzzy Logic Controller For Power System Stabilization," *Electric Machines \& Power Systems*, vol. 25, no. 4, pp. 429-441, 1997.

[3] A Demir et al., "Modeling and control of a nonlinear half-vehicle suspension system: A hybrid fuzzy logic approach," *Nonlinear Dynamics*, vol. 67, no. 3, pp. 2139-2151, 2012.

[4] C.-H. T. H.-C. Lu, "Design and Implementation of a DSP-Based Grey-Fuzzy Controller for Induction Motor Drive," *Electric Machines \& Power Systems*, vol. 28, no. 4, pp. 373-384, 2000.

[5] W. Chen, H. mei Yuan and Y. Wang, "Design and implementation of Digital Fuzzy-PID controller based on FPGA," in *Proc. 4th IEEE Conf. Industrial Electronics and Applications ICIEA 2009*, 2009.

[6] "Blackfin Embedded Processor - ADSP-BF537," 2010.

[7] E. Frias-Martinez, "Real-Time Fuzzy Processor on a DSP," in *Proceedings. 8th IEEE International Conference on Emerging Technologies and Factory Automation, 2001.*, 2001.

[8] S. Jung and S. su Kim, "Hardware Implementation of a Real-Time Neural Network Controller With a DSP and an FPGA for Nonlinear Systems," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 265-271, 2007.

[9] S. Ahn, Y. H. Kim and W. H. Kwon, "Design of a fuzzy logic controller module for a loop controller," in *Proc. IEEE Int Industrial Technology (ICIT '96) Conf*, 1996.

[10] C.-S. Chen and W.-L. Chen, "Robust fuzzy controller design of nonlinear dynamic systems based on H âˆž optimal criterion," *International Journal of Systems Science*, vol. 33, no. 12, pp. 1013-1023, 2002.

[11] F.-H. Hsiao, S.-D. Xu, C.-Y. Lin and Z.-R. Tsai, "Robustness Design of Fuzzy Control for Nonlinear Multiple Time-Delay Large-Scale Systems via Neural-Network-Based Approach," *{IEEE} Trans. Syst., Man, Cybern. {B}*, vol. 38, no. 1, pp. 244-251, 2008.

[12] S. M. Halpin and R. F. I. Burch, "Applicability of neural networks to industrial and commercial power systems: a tutorial overview," in *Proc. Industrial and Commercial Power Systems Technical Conf. Conf. Record, Papers Presented at the 1996 Annual Meeting. IEEE 1996*, 1996.

[13] J.-S. R. Jang and C.-T. Sun, "Neuro-fuzzy modeling and control," *Proc. {IEEE}*, vol. 83, no. 3, pp. 378-406, 1995.

[14] T. Lie, G. Shrestha and A. Ghosh, "Design and application of a fuzzy logic control scheme for transient stability enhancement in power systems," *Electric Power Systems Research*, vol. 33, no. 1, pp. 17-23, 1995.

[15] M. K. Panda, G. N. Pillai and V. Kumar, "Design of an Interval Type-2 Fuzzy Logic Controller for Automatic Voltage Regulator System," *Electric Power Components and Systems*, vol. 40, no. 2, pp. 219-235, 2011.