Low-power Fuzzy Logic VLSI Implementation with Asynchronous Topology for Neuronal Sensors

B. I. Morshed*, S. Consul-Pacareu
The Department of Electrical and Computer Engineering
The University of Memphis
Memphis TN USA 38152
Email: bmorshed@memphis.edu*

***Abstract*—Embedded systems for ubiquitous sensing towards the next-generation cyber-physical systems require low-power design approaches. A non-traditional low-power asynchronous circuit design for Fuzzy Logic rule-block is presented in this paper. The developed low-power architecture of the Fuzzy rule blocks consumes 197.2 μW for 3 rules using CMOS 0.13 micron technology. Implementation with an asynchronous topology reduced the power consumption to 64.5 μW. Such low-power controllers would be attractive for embedded neuronal sensors powered by energy scavenging.**

I. Introduction

A Fuzzy Logic controller is based on "If *condition*, then *action*" based rules. Non-linear controllers based on Fuzzy Logic are stable and suited for cyber-physical systems. Human conceivable rule-based operations, high-adaptability and reconfigurability of Fuzzy Logic makes it an attractive choice for robust, adaptive, sustainable, and reconfigurable systems. To take full advantage of Fuzzy Logic, it needs to be implemented in hardware with efficient, low-power design topologies. Such scheme would be very attractive for battery-less embedded neuronal sensors that are powered by energy scavenging.

VLSI implementation of Fuzzy Logic functions can be operated with a very low-power [1]. The power dissipation using CMOS 0.35μ technology was 150 μW per rule with the total delay time of 200 ns and maximum operation error 1%. In this paper, a revised design of Fuzzy Logic implementation is reported with an asynchronous topology that significantly reduces the power requirement to a fraction of the original.

II. Circuit Implementation of Fuzzy Logic

The circuit implementation of Fuzzy Logic block shown in Fig. 1 is designed using Virtuoso Analog Design Eenvironment (Cadence Inc., VA, USA). A rule block takes an analog input (crisp input) and converts it using Fuzzifier, which consists of input membership functions. A Fuzzy reasoning with a rule base combines the inputs to produce an output and is usually realized by Min/Max circuits. A Defuzzifier combines the Fuzzy outputs, calculates the weighted average and produces a analog output (crisp output). In this implementation, the inputs and outputs are in voltage-mode to provide parallel operability and large fan-out. The internal signals are in current-mode, which provides MOS threshold voltage independent operation.

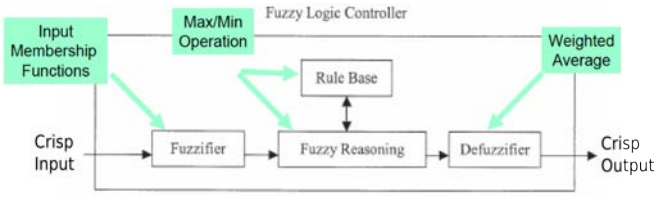

Fig. 1. Various blocks of a Fuzzy Logic controller.

A. Input membership function

A circuit level implementation of a 3-input Gaussian membership function generator is given in Fig. 2. The circuit takes input as voltage and produces output as current. A bias voltage (V_b) is given to the circuit which determines the maximum current level. The bias voltage is generated by a separate bias circuit. A unit voltage (V_u) is given to the bias circuit which normalizes the input voltages. The output of the input membership function generator is given in Fig. 3. The intersecting points of the curves are defined by the control input voltages ($V_{c1} - V_{c4}$) in Fig. 2.

B. Maximum and minimum circuits

The maximum circuit operates in similar fashion of "winner takes all" circuit. A positive feedback mechanism ensures that the circuit output closely follows the maximum input. For minimum circuit, a binary tree circuit is implemented that takes two current inputs and the output follows the minimum of the two input currents. The transient responses of a 2-input maximum operator circuit and a 2-input minimum operator circuit are given in Fig. 4. The outputs verify the desired functionalities. Designing Min/Max circuits with higher number of inputs are trivial.

C. Weighted average circuit

A weighted average circuit has been designed to produce voltage as output signal while taking current as input signal as shown in Fig. 5. The two pairs of FETs on the left side operates in bounded-difference mode and compares the input to a reference signal level (I_{ref}). They are converted to voltage levels and passed to a push-pull type configuration of FETs, which fights over a mid-level voltage output produced by a simple register based voltage divider. This can be simply replaced by an equivalent circuit producing $V_{dd}/2$.

978-1-4673-1140-3/12/$26.00 ©2011 IEEE

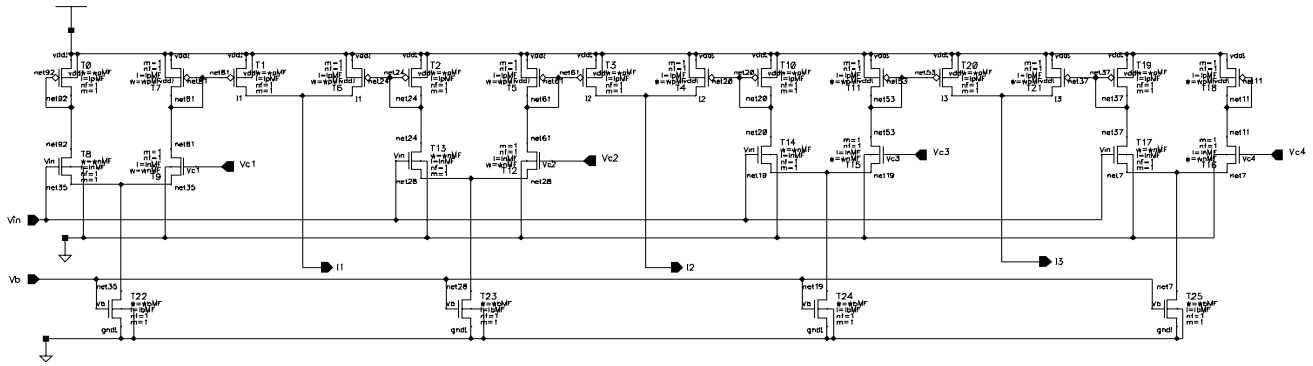

Fig. 2. The circuit diagram to produce a 3 Gaussian membership functions for the Fuzzifier block.

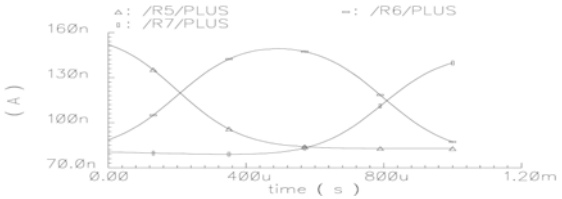

Fig. 3. The transient output of the Gaussian membership functions.

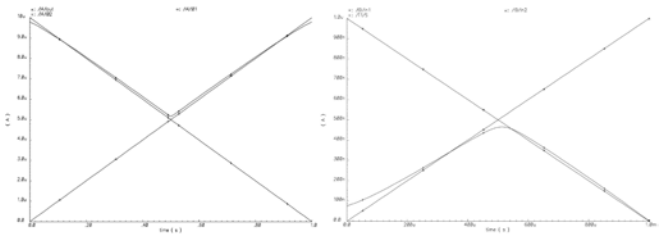

Fig. 4. Transient responses of the 2-input maximum and minimum operators.

III. Implementation with Asynchronous Topology

Asynchronous implementation is done with the merge-element to be high during the transition of signal through the delay element. Designs for C-element and merge-element are given in Fig. 6 and Fig. 7, respectively. Functional verification was successful (not shown due to space limit).

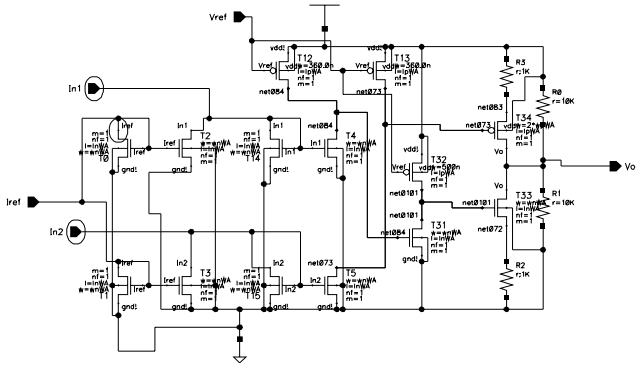

Fig. 5. A weighted average circuit.

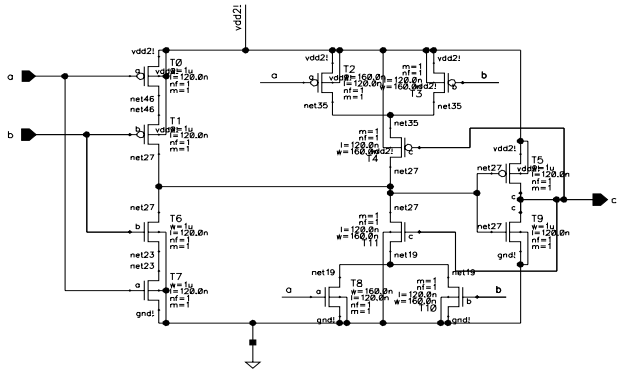

Fig. 6. A C-element circuit.

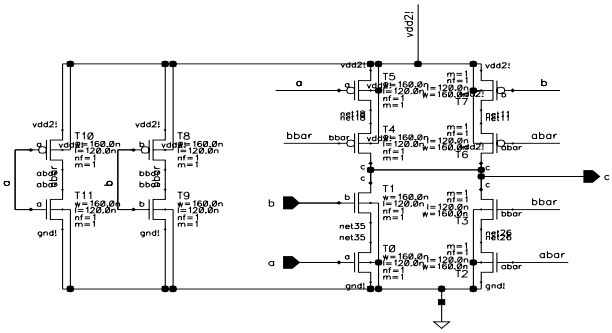

Fig. 7. A merge-element circuit.

IV. Results and Conclusion

Using 0.13μ technology, the power consumed by the Fuzzy block is 197.2 μW that contained 3 rules. Average power consumption per rule is 65.7 μW. With the incorporation of asynchronous topology, power consumption reduced to 64.5 μW; an average of 21.5 μW per rule. This significant amount of power reduction ($\sim$67%) is achieved at the expense of higher area requirement and increased complexity.

References

[1] M. Sasaki, T. Inoue, Y. Shirai, F. Ueno, "Fuzzy multiple-input maximum and minimum circuits in current mode and their analyses using bounded-difference equations", *IEEE Trans. Computers*, vol. 39, no. 6, pp. 768 - 774, 1990.