A 1.6V analog integrated implementation of a Gaussian Mixture Model classifier

Emmanouil Anastasios Serlis, National Technical University of Athens, Greece, School of Electrical and Computer Engineering, manosserlis@gmail.com

***Abstract*- This paper presents the implementation of a 1.6V analog Gaussian Mixture Model (GMM) classifier, which includes a Gaussian circuit and a Winner-Take-All (WTA) circuit. The circuit characteristics as well as the functionality of the entire classification system are analytically presented. The electronics analysis was done in the Cadence IC Suite simulation software. Direct comparison of the hardware implementation with its software counterpart is presented in a test thyroid dataset to prove the validity of the analog design.**

Index Terms – Embedded Machine Learning, Analog VLSI, Classifier, Gaussian Circuits

I. INTRODUCTION

The unprecedented increase in data availability and computing power has given rise to the use of machine learning algorithms and techniques, to directly solve a plethora of problems, from medical prediction system [1] to fault diagnosis [2]. Under that premise, the demand for hardware that can efficiently execute such algorithms has risen, with both digital and analog implementations being presented. Regarding the digital ones, accelerating systems based on Field Programmable Gate Arrays (FPGAs) [3] and digital ASICs [4] offer a solid balance of power consumption, computing speed and consumed chip area. However, their analog counterparts can offer an even larger benefit in all of the areas mentioned above, due to their capability of sub-threshold operation, which significantly reduces the power consumption needs.

Based on all the aforementioned, this work presents the implementation of a fully analog integrated Gaussian Mixture Model (GMM) classifier. At first, the main GMM premises and principles, both as a mathematical model and a hardware implementation, will be discussed. Next, we present a thorough analysis of the main building blocks of the classifier, i.e. the Gaussian Bump circuit-where three different implementations will be analyzed-and the Winner-Take-All (WTA) circuit. Finally, an exhibition of the hardware system classification capabilities will be presented and compared with a software-based one.

II. ANALYSIS OF THE GMM CLASSIFIER

A GMM is a probabilistic model which assumes that all the data points are generated from a finite number of Gaussian distributions with unknown parameters. Usually, GMMs are used in an unsupervised manner, where the labels of the input data are not known, thus leading to solving clustering problems. However, regarding classification problems, a single GMM could be used for data clustering of one class, independently from the other ones. For each of the C classes, the posterior probability of each class is given via the Bayes theorem as:

$$P(y_c \mid x) = \frac{P(x|y_c)P(y_c)}{P(x)} \quad (1)$$

,where $P(y_c)$ is the prior probability and P(x) the evidence probability of the input vector x. Implementing eq. (1) for all of the c distinctive classes, the winning class is the one with the maximum posterior probability, or:

$$y_{out} = argmax[P(y_c)P(x \mid y_c)] \quad (2)$$

In terms of the hardware implementation of the model, a mock example for 3 Classes and 5 input features is displayed in Fig 1. In particular, the proposed system consists of three different cluster cells, one for each different class. Inside each cluster cell, 5 bump circuits are connected, where the output current of bump circuit i is used as input bias current for bump circuit $i+1$. All 5

bumps inside a cluster cell have tunable electronic components regarding the mean value V_{mean} and the variance V_c of the circuit, which have been defined via model software training.

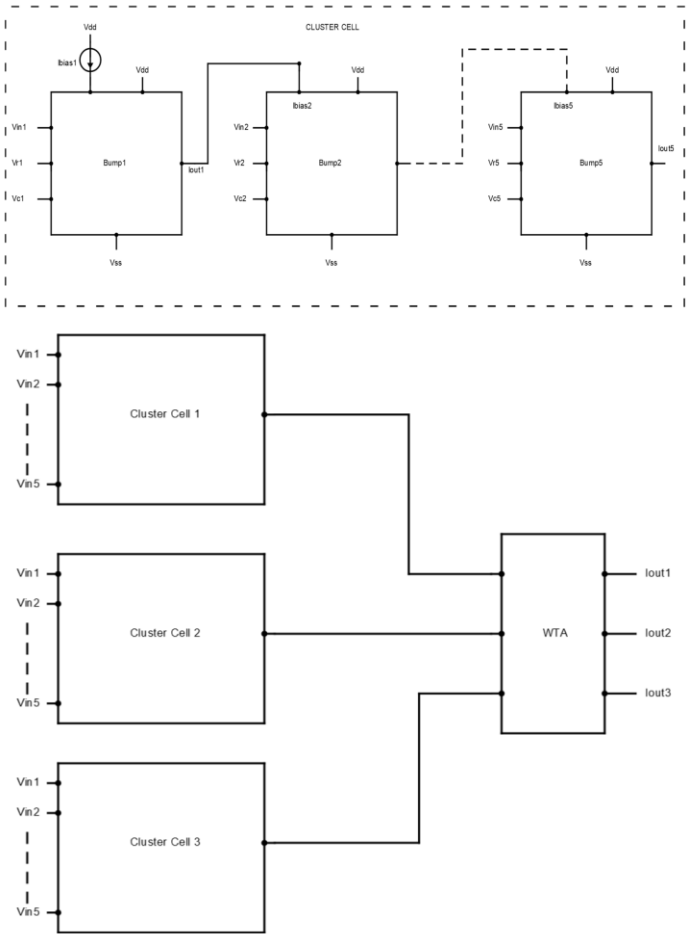

Fig. 1: Diagram Analysis of the proposed GMM Classifier (a) A cluster cell used for classification of a 5-feature dataset, where the output current of each Bump circuit is used as input for the next one (b) Whole-System Architecture, based on the cluster cell of Fig 1a and a WTA circuit

The input features are given in voltage values on each cluster cell, which produces an output current equivalent to the posterior probability of eq. 1. Finally, all output cluster currents are being utilized as inputs on the WTA circuit, which resembles the argmax function of eq 2. The cluster current with the highest value is being given the entire of the WTA's current, while the rest of the classes' currents are being reduced to zero.

III. CIRCUIT ANALYSIS

In this section, three different circuit implementations for the Gaussian function and one for the argmax operation will be presented. A direct comparison of all three different bump architectures will be implemented, for the selection of one of them, in order to be used in the whole system architecture. Power supply for all circuits is set at $V_{DD} = 1.6V \; and \; V_{SS} = 0V$. Current bias values for circuitry analysis are set at $I_{bias} = 16nA$

A. Fully Tunable, bulk controlled Bump

The initial circuit shown in Fig.2 is a modification of the first Gaussian function circuit, introduced by Delbruck in 1991[5]. It consists of a current mirror (M1-M2) a cascode differential block (M3-M6) and a non-symmetric current correlator (M7-M10). In contrast, to the initial implementation by Delbruck, M3-M4 transistors have been added in the differential pair with tunable body voltage, in order to modify the Gaussian variance accordingly.

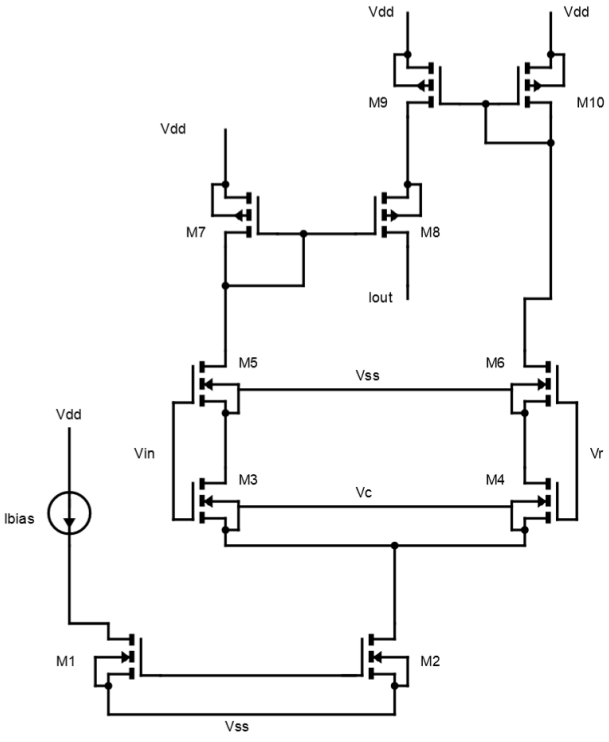

Fig. 2: Fully tunable, bulk-controlled modification of Delbruck's bump.

Transistor Name	W(um)/L(um)
M1	7.5/1.2
M2	15/1.2
M3-M6	7.5/1.2
M7	15/1.2
M8-M9	7.5/1.2

M10	15/1.2

Table 1: MOS Transistor Sizes for Modified Delbruck's Bump

B. *Gilbert's Gaussian Circuit*

In contrast to Delbruck's Gaussian Circuits, there is also the design principle of using multiple differential pairs that add or subtract circuits. One such example is Gilbert's Gaussian Circuit that consists of three differential pairs and is based on Gilbert's multiplier [6]. The main disadvantage of Gilbert's bump shown in Fig. 3 is the lack of electronic control over the variance, thus making it inappropriate to use for high-accuracy hardware classification results.

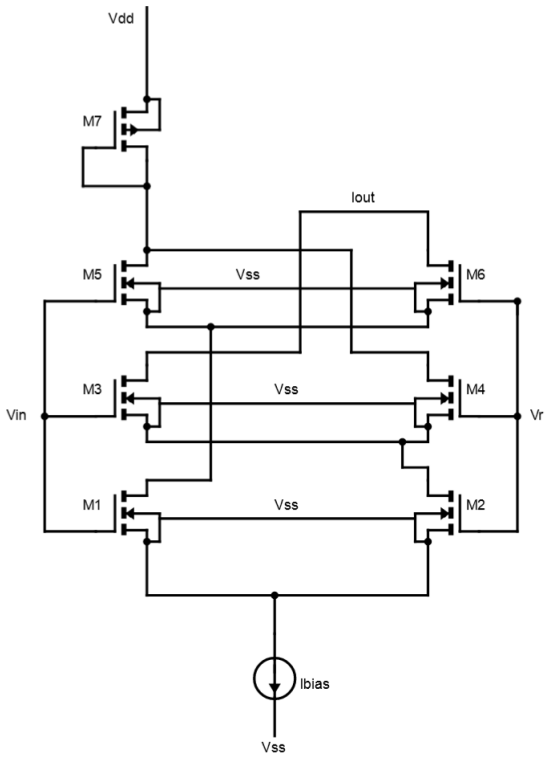

Fig. 3: Gilbert's Gaussian Circuit

Transistor Name	W(um)/L(um)
M1-M2	16/4.05
M3-M4	16/4.05
M5-M6	8/4.05
M7	4/4.05

Table 2: MOS Transistor Sizes for Gilbert's Gaussian Circuit

C. *Modified Bump Circuit*

In comparison to the Delbruck's Bump Circuits that use non-symmetrical current correlator blocks, a modified bump circuit with symmetrical current correlator is presented in Fig 3. In that way, a more symmetrical Gaussian function even for small bias currents can be achieved, as shown in [7]. Furthermore, the cascode current mirror consisting of transistors M1-M6 is used to increase mirroring effect even for small bias currents, thus leading to a more robust current Gaussian function and allowing higher training parameter variability.

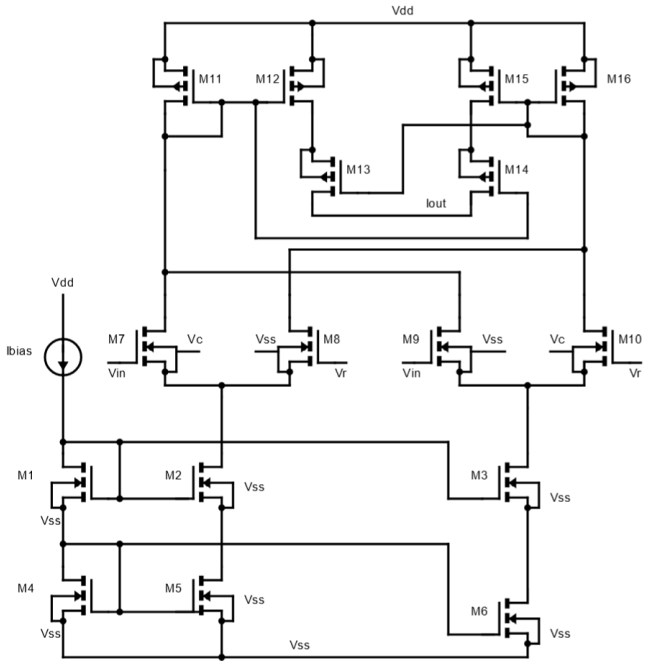

Fig. 4: Modified Bump Circuit

Transistor Name	W(um)/L(um)
M1-M4	$w_{mirr}/2$
M5-M6	$4 * w_{mirr}/2$
M7	$2*w_{diff}$ /2
M8-M9	w_{diff} /2
M10	$2*w_{diff}$ /2
M11	$4*w_{corr}/2$
M12-M15	$w_{corr}//2$
M16	$4*w_{corr}//2$
w_{mirr}=w_{diff}	64um
w_{corr}	10um

Table 3: MOS Transistor Sizes for Modified Bump Circuit

D. Noise and PSRR Comparison of all three Gaussian Circuits

In this section, noise and PSRR specification results for the aforementioned MOS dimensions and power supply values are presented for all 3 proposed Gaussian circuits. As we can clearly notice in figures 5 and 6, the modified bump proposed in [7] offers the lowest noise values, especially in low to mid frequencies, as well as the second highest low-frequency PSRR values in absolute. Based on all the above, the modified Bump circuit will be used in the whole-system training in Section IV.

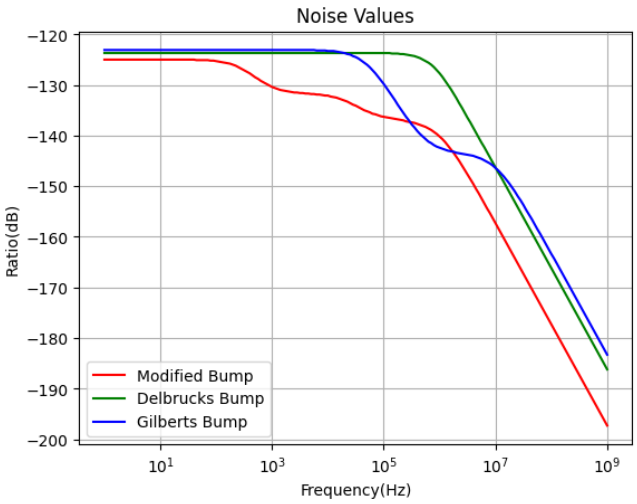

Fig. 5: Noise values for all Gaussian Circuits

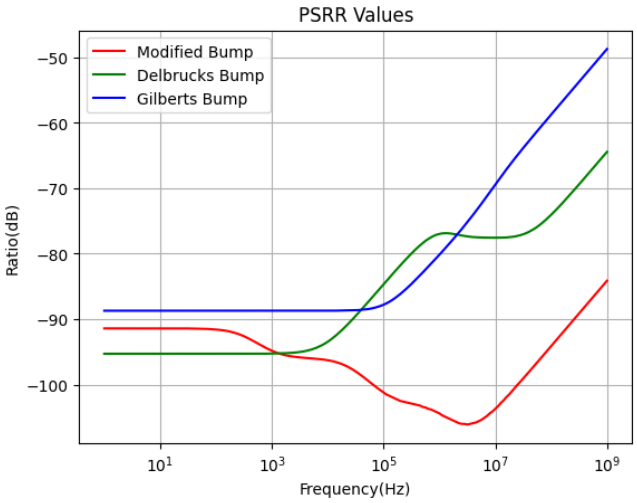

Fig. 6: PSRR values for all Gaussian Circuits

Spec Name	Initial Target	Delbruck	Gilbert	Modified
VDD	[1.6V,2.4]	1.6V	1.6V	1.6V
IBIAS	<10uA	16nA	16nA	16nA
PSRR(1Hz)	>35dB	-95.2	-88.7	-91.4
PSRR(1kHz)	>12dB	-95.2	-88.7	-91.4
Noise(10Hz)	<-92dB	-123.6	-123.0	-124.9
Noise(100Hz)	<-100dB	-123.6	-123.0	-125.2
Noise(1kHz)	<-106dB	-123.6	-123.0	-130.3
Noise(10kHz)	<-110dB	-123.6	-123.2	-132
Noise(100kHz)	-	-123.7	-129.8	-136.3
Noise(1MHz)	<-140dB	-127	-142.4	-140.1

Table 4: Gaussian Circuits' Specifications

E. Parametric Analysis for Modified Bump Circuit

Regarding the modified bump circuit, a significant aspect refers to the optimization of sizing parameters presented in table 3, in order to receive reproducible Gaussian functions for a wide range of electronic mean and variance values. In that way, lower validation errors can be achieved, due to a better correlation between software and hardware training parameters.

In terms of sizing, parametric analysis for w_{mirr}, w_{diff} and w_{corr} are presented in figures 7-9. It is shown that an increased transistor width in the current mirror results in the output current receiving values closer to I_{bias}, which is a desirable feature for achieving reproducible Gaussian height. Moreover, an increase in the width of the differential pair is inversely proportional to the gaussian variance. Even though a low gaussian variance could result in sub-optimal simulation results, that is also fine-tuned by voltage Vc in transistors M7 and M10.

Finally, w_{corr} is a highly sensitive parameter since an increase to values higher than 20um corresponds to output current values higher than I_{bias}. From the analysis above, sizing parameters of w_{mirr}=w_{diff}=64um and w_{corr} = 10um were chosen, as in Table 4.

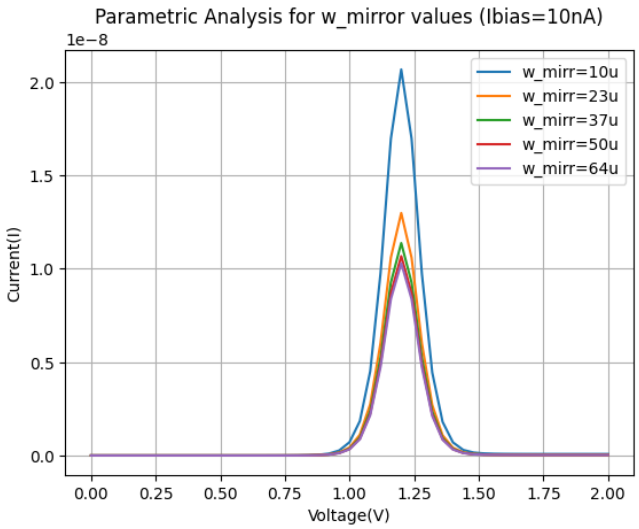

Fig. 7: Parametric analysis for different w_{mirr} values

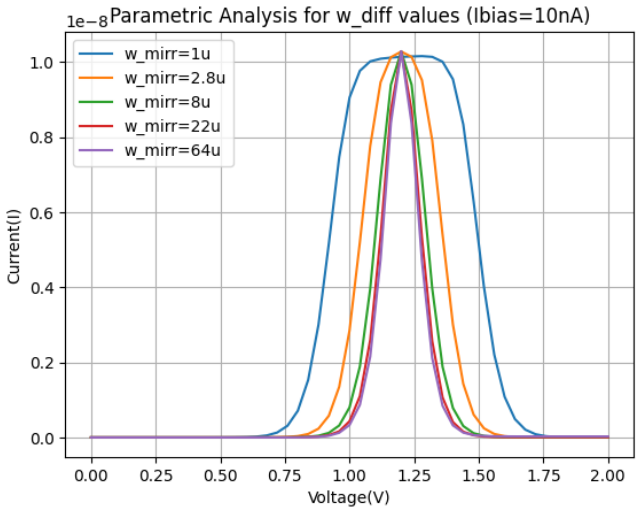

Fig. 8: Parametric analysis for different w_{diff} values

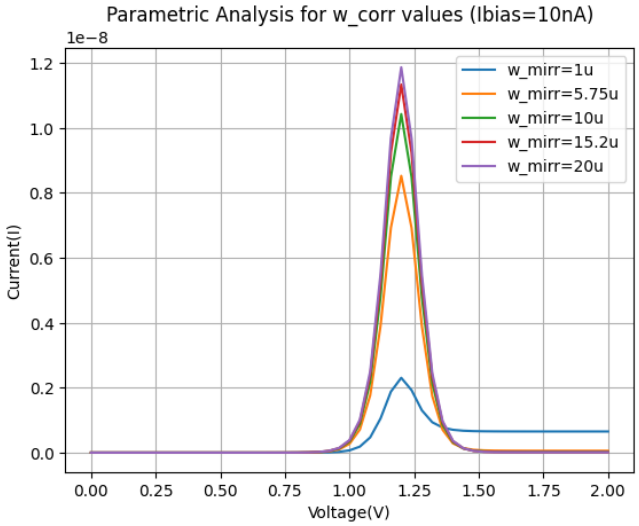

Fig. 9: Parametric analysis for different w_{corr} values

After defining sizing parameters for constant mean and variance values ($V_r = 0.8V$ and $V_c = 0V$), another aspect of high significance has to do with the range of mean and variance voltage values. Results are shown in figures 10-11, which indicate that Vr values of [0.7V, 1.1V] and Vc values of [0V, 0.4V] would lead to an adequate range of training parameters, while not distorting the Gaussian PDF produced by the modified circuit.

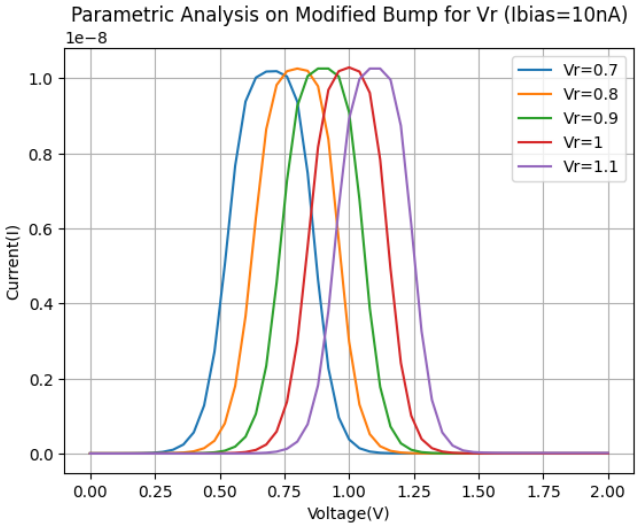

Fig. 10: Parametric analysis for different Vr values

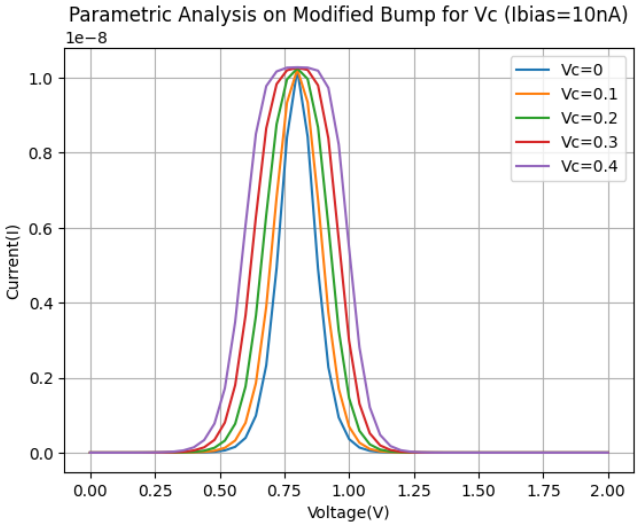

Fig. 11: Parametric analysis for different Vc values

F. Lazzaro WTA Circuit

A typical Lazarro WTA circuit [8] was used to determine the winning class and, thus, simulate the argmax function in hardware. The WTA circuit

displayed in Fig. 12 consists of C=3 NMOS neuron cells, where C is the number of discrete classes, with a common bias current for all neurons. Each neuron is responsible for the input and output of a single class. In particular, the output current of the neuron with the largest input current has a non-zero value, while the rest are zero. As we can notice in the results of Fig. 13, the NMOS WTA can switch the winning class from I2 to I1 at the correct I1 current (30 nA). However, it should be stated that the switch of the winning class is done in a linear fashion and not in a switch-like one, a fact that partially explains the accuracy drop in Section IV.

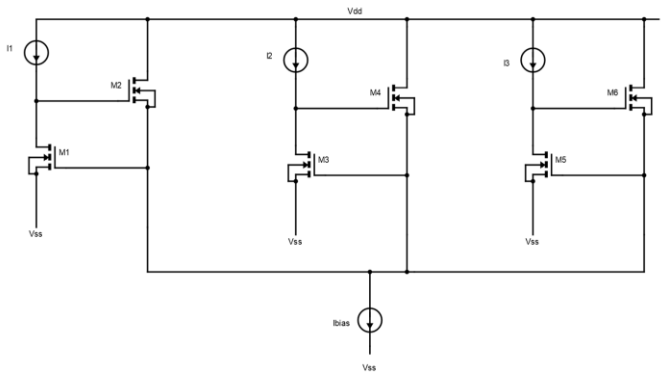

Fig. 12: NMOS WTA Circuit

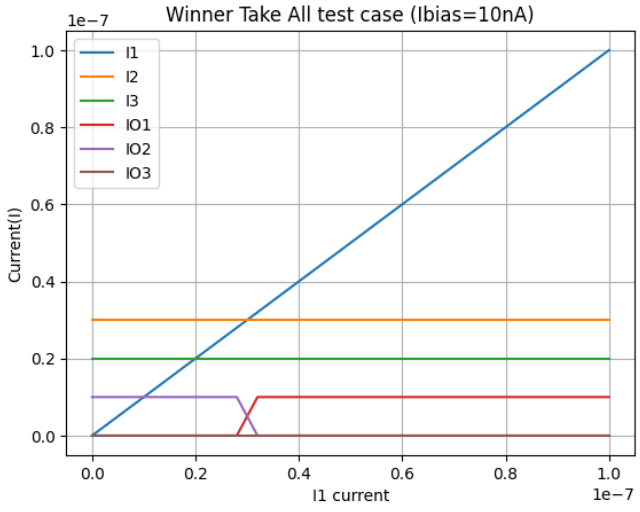

Fig. 13: Mock example of 3 WTA use for 3 input classes

Transistor Name	W(um)/L(um)
M1-M6	6/2

Table 5: MOS Transistor Sizes for 3-class NMOS WTA Circuit.

IV. SYSTEM VALIDATION

In order to examine the validation of the system presented in section II and the circuits designed and analysed in section III a mock dataset was used and compared to a software implementation. The dataset chosen was the Thyroid Disease Data Set[9] which includes 5 features and 4 classes. The hardware system used is that of figure 1, both for the cluster cell as well as for the whole GMM system.

System training was implemented offline in Python Software (via the Sklearn library). As a result, I_{bias}, Vr and Vc values were received via software training and then used in the hardware simulations executed in the Cadence IC Suite package. Table 6 shows the minimum and maximum values chosen for training parameters of bump I_{bias}, Vr and Vc. Total simulation time in Cadence Suite was set at 6.5msec (or 0.1 msec classification period for each of the 65 test cases).

Training Parameter	Min Value	Max Value
Ibias	16nA	16nA
Vr	0.8V	1.05V
Vc	0V	0.5V

Table 6: Minimum and Maximum Parameters for software training

Results of hardware training and validation implementations are shown in figures 14-16, as well as in table 7. It has been proven that the classifier provides high-level accuracy results (a 5% drop in validation accuracy was noticed) and a total power consumption lower than 10uW, that was set as a maximum threshold specification. For more in depth results of the software and hardware implementations, confusion matrix and classification report figures are provided in the Appendix section.

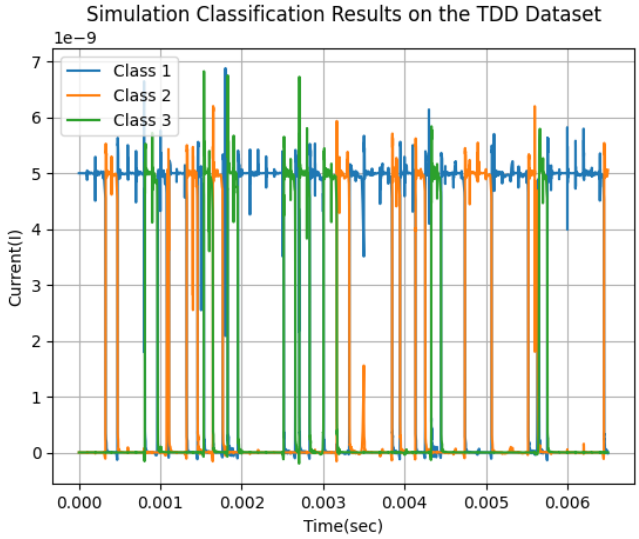

Fig. 14: Simulation Classification Results on the TDD Dataset

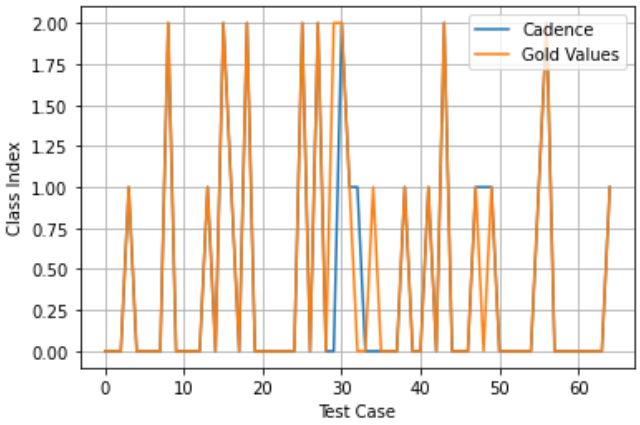

Fig. 15: Cadence classification results compared to the correct dataset labels

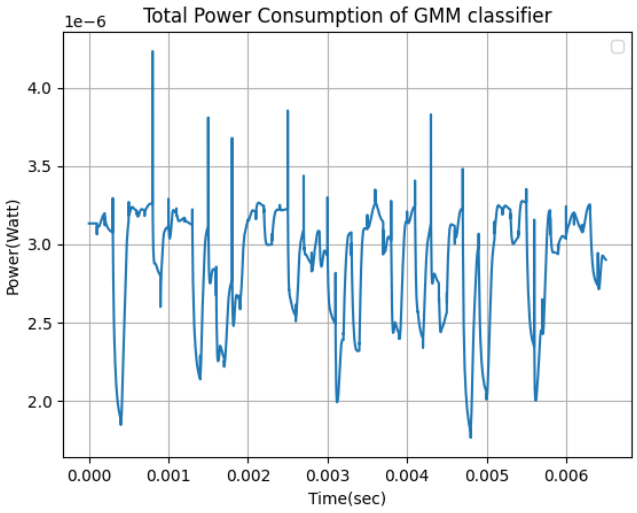

Fig. 16: Total power consumption of the GMM Classifier used in the TDD dataset.

	Software	Hardware
Validation Accuracy	98.4%	93.8%

Table 7 : Direct Comparison of Software and Hardware validation accuracies.

V. CONCLUSION

In this work, an architecture for tunable analog integrated GMM-based classifiers was introduced and analyzed By modifying and using Gaussian function and WTA circuits, GMM-based classifiers targeting problems with various numbers of classes, clusters and data dimensionalities can be implemented. The proposed architecture was applied on a thyroid diseases' test dataset. Its parameters were generated through offline training of a GMM classifier in software. Extensive analysis and comparisons of the classification results, on these problems, highlight the proper operation of the proposed architecture and justify its utility as a low-power, high-efficiency circuit implementation of the GMM. Such type of workflow is easily reproduced for other type of classification systems and problems, such as object recognition[10] and image classification[11].

REFERENCES

[1] N. G. Maity and S. Das, "Machine learning for improved diagnosis and prognosis in healthcare," in *2017 IEEE Aerospace Conference*, Big Sky, MT, USA, Mar. 2017, pp. 1–9. doi: 10.1109/AERO.2017.7943950.

[2] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mech. Syst. Signal Process.*, vol. 138, p. 106587, Apr. 2020, doi: 10.1016/j.ymssp.2019.106587.

[3] S. Branco, A. G. Ferreira, and J. Cabral, "Machine Learning in Resource-Scarce Embedded Systems, FPGAs, and End-Devices: A Survey," *Electronics*, vol. 8, no. 11, p. 1289, Nov. 2019, doi: 10.3390/electronics8111289.

[4] V. J. Sorger, "Photonic devices, ASICs and systems for machine intelligence," in *Active Photonic Platforms (APP) 2022*, San Diego, United States, Oct. 2022, p. 56. doi: 10.1117/12.2632659.

[5] T. Delbruck, "'Bump' circuits for computing similarity and dissimilarity of analog voltages," in *IJCNN-91-Seattle International Joint Conference on Neural Networks*, Seattle, WA, USA, 1991, vol. i, pp. 475–479. doi: 10.1109/IJCNN.1991.155225.

[6] B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response," *IEEE J. Solid-State Circuits*, vol. 3, no. 4, pp. 365–373, Dec. 1968, doi: 10.1109/JSSC.1968.1049925.

[7] V. Alimisis, G. Gennis, K. Touloupas, C. Dimas, M. Gourdouparis, and P. P. Sotiriadis, "Gaussian Mixture Model classifier analog integrated low-power implementation with applications in fault management detection," *Microelectron. J.*, vol. 126, p. 105510, Aug. 2022, doi: 10.1016/j.mejo.2022.105510.

[8] K. Urahama and T. Nagao, "K-winners-take-all circuit with O(N) complexity," *IEEE Trans. Neural Netw.*, vol. 6, no. 3, pp. 776–778, May 1995, doi: 10.1109/72.377986.

[9] "Thyroid Disease Data Set." Jan. 01, 1987. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/thyroid+disease

[10] R. Genov and G. Cauwenberghs, "Kerneltron: support vector 'machine' in silicon," *IEEE Trans. Neural Netw.*, vol. 14, no. 5, pp. 1426–1434, Sep. 2003, doi: 10.1109/TNN.2003.816345.
[11] R. Zhang and T. Shibata, "An analog on-line-learning K-means processor employing fully parallel self-converging circuitry," *Analog Integr. Circuits Signal Process.*, vol. 75, no. 2, pp. 267–277, May 2013, doi: 10.1007/s10470-012-9980-y.

VI. APPENDIX

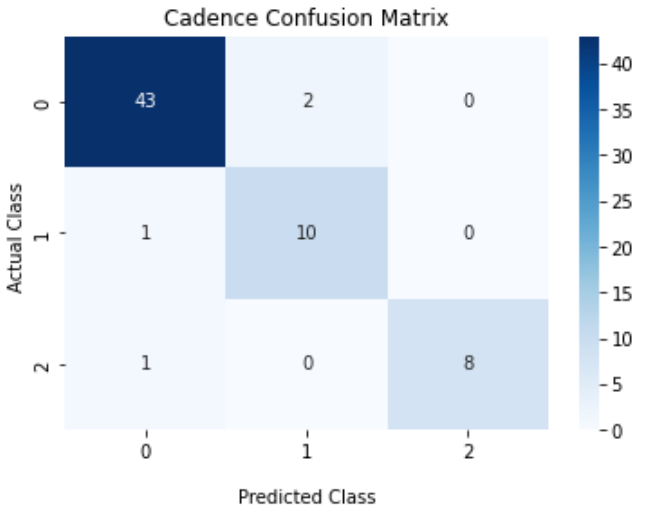

Fig. 17: Cadence Confusion Matrix

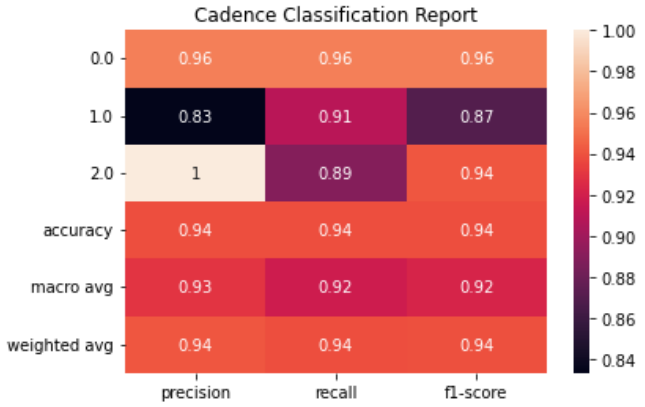

Fig. 18: Cadence Classification Report

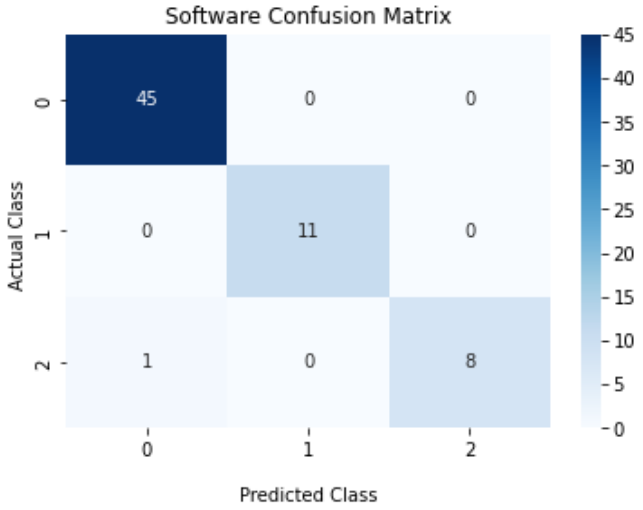

Fig. 19: Software Confusion Matrix

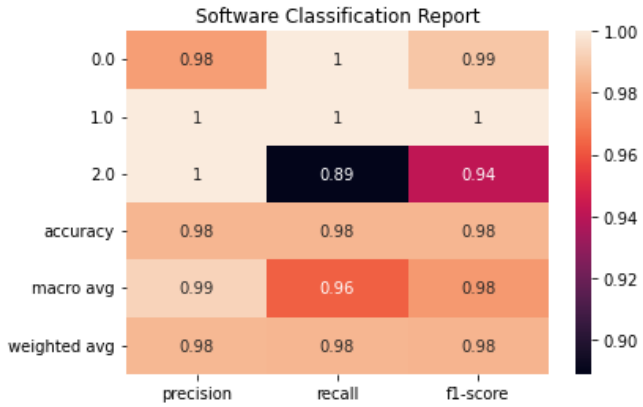

Fig. 20: Software Classification Report

VII. AUTHORS

First & Correspondence Author

Emmanouil Anastasios Serlis was born in Athens, Greece, in 2000 and is currently an undergraduate student in the School of Electrical & Computer Engineering of National Technical University of Athens. (manosserlis@gmail.com ,+30 6943096574)