Received October 29, 2020, accepted November 9, 2020, date of publication November 16, 2020,
date of current version November 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3038052

Microelectronic CMOS Implementation of a Machine Learning Technique for Sensor Calibration

JAVIER A. MARTÍNEZ-NIETO[1], MARIA TERESA SANZ-PASCUAL[2], (Member, IEEE), N. MEDRANO[1], (Senior Member, IEEE), BELEN CALVO LOPEZ[1], (Senior Member, IEEE), AND DIEGO ANTOLÍN CAÑADA[1]

[1]Group of Electronic Design (GDE), University of Zaragoza, 50009 Zaragoza, Spain
[2]Electronics Department, National Institute of Astrophysics, Optics and Electronics (INAOE), Puebla 72840, Mexico

Corresponding author: Javier A. Martínez-Nieto (almartinez@unizar.es)

This work has been supported in part by the National Council of Science and Technology (CONACYT) through the CB-2015-257985 Research Project, and in part by the Ministerio de Ciencia e Innovación, through the PID2019-106570RB-I00 / AEI / 10.13039/501100011033 Research Project.

ABSTRACT An integrated machine-learning based adaptive circuit for sensor calibration implemented in standard 0.18μm CMOS technology with 1.8V power supply is presented in this paper. In addition to linearizing the device response, the proposed system is also capable to correct offset and gain errors. The building blocks conforming the adaptive system are designed and experimentally characterized to generate numerical high-level models which are used to verify the proper performance of each analog block within a defined multilayer perceptron architecture. The network weights, obtained from the learning phase, are stored in a microcontroller EEPROM memory, and then loaded into each of the registers of the proposed integrated prototype. In order to verify the proposed system performance, the non-linear characteristic of a thermistor is compensated as an application example, achieving a relative error e_r below 3% within an input span of 130°C, which is almost 6 times less than the uncorrected response. The power consumption of the whole system is 1.4mW and it has an active area of 0.86mm^2. The digital programmability of the network weights provides flexibility when a sensor change is required.

INDEX TERMS Adaptive signal processing, artificial neural networks, CMOS, sensor conditioning.

I. INTRODUCTION

In sensor production it is desired that all the sensors have the same well-defined characteristic with a certain accuracy. However, due to process variations, properties vary from device to device and so do transfer characteristics. Offset, gain, and non-linearity errors are the most common errors arising in the transfer function when the sensor is characterized.

Therefore, in order to produce reliable sensors, it is necessary to correct or minimize these errors, and consequently a correction or calibration process is required. This process consists in the standardization of the device response so that it matches an expected function, in such a way that all the sensors of the same kind, regardless of the batch, always present the same characteristic.

The associate editor coordinating the review of this manuscript and approving it for publication was Aijun Yang.

Different techniques can be used to correct the different sensor transfer errors. The calibration systems can be fully analog. This approach mainly focus in linearizing the sensor response by using analog circuits to generate a characteristic that approximates the inverse of the sensor behavior [1], [2]. These techniques are the simplest and have least cost in terms of area and power consumption, however they lack of flexibility when a different type of sensor is employed.

Digital techniques offer more flexibility and accuracy, but the circuit complexity and processing time are higher [3]. Tipically, the calibration is carried out after the A/D conversion, so high resolution converters are required. The most common digital technique is the use of look-up tables (LUT), which allow to correct any non-linearity, but the memory requirements and the number of calibration points are high [4]. Other techniques are based on piecewise-linear (PWL) approach [5], or progressive polynomial method [6]. These techniques require less calibration points and memory, but

it is necessary to know the type of non-linearity in order to properly select the calibration points and the number of correction coefficients.

The calibration can also be performed in the conversion step using non-linear ADC converters [7]–[9], so that both, the digital conversion and linearization are performed simultaneously. These techniques minimize digital implementation costs, but have no flexibility when using different sensors.

Compared with these techniques, machine learning (ML) based techniques [10] have the advantage that they can linearize different functions without having a prior knowledge of the particular sensor non-linearity, providing flexibility to calibrate different types of devices [11]–[13]. Furthermore, compared with LUTs, these techniques require less calibration points and the memory requirements are also lower. If besides, a mixed-mode solution (analog processing signal with digital programmability) is adopted, we can take advantage of the low-voltage low-power characteristics of analog processing electronics, as well as the digital programmability of register-based structures, thus lending great flexibility to the system.

In particular, an optimal solution is a mixed-mode integrated ML model with analog processor units, to minimize power consumption and area, and digital programmability, to facilitate the reprogramming of the model parameters. Therefore, this approach constitutes a flexible solution at a low cost in terms of area, power consumption and computational complexity, thus being a valuable choice for adaptive sensor processing in embedded applications [14]–[16]. Furthermore, it is worth mentioning that for the realization of these systems, CMOS is the most suitable technology due to the cointegration capability of sensors and electronics required for the conditioning, thus achieving low cost compact systems.

This paper presents an integrated CMOS mixed-mode machine learning model based on a multilayer perceptron (MLP) configuration, thus providing an efficient and robust method to compensate any kind of non-linear output sensor response. A description of the proposed CMOS building blocks conforming the mixed-mode MLP processing unit is presented in Section II, where an experimental characterization of each circuit is also presented. Section III shows a summary of the methodology described in [17] to efficiently carry out the ML-based system high-level simulations. The proposed system electrical characterization considering a real example of non-linear sensor characteristic and a performance comparison with related and similar works are presented in Section IV. Finally, conclusions are drawn in Section V.

II. INTEGRATED PROCESSOR BUILDING BLOCKS

The main processing unit in a machine learning systems is the neuron, whose general structure is shown in Figure 1. It consists of three building blocks: the multipliers, which multiply either the input or the intermediate layer signals by a set of coefficients that weigh the contribution of each input in the processor output; the adder to sum the input weighted

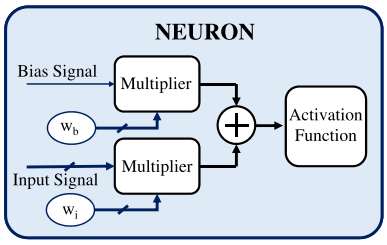

FIGURE 1. Basic processing unit in ML-based systems.

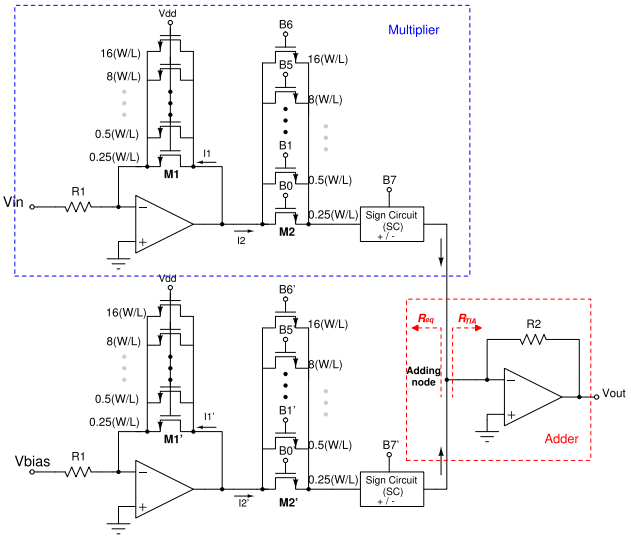

FIGURE 2. 8-bit analog-digital multiplier circuit and addition node.

signals plus an additional input called bias, and the activation function (AF) circuit that implements the non-linear operation with the previous weighted sum and generates the neuron output. The electrical implementation of each processor building block in standard 0.18μm CMOS technology with $1.8V$ power supply has been studied in [17], where electrical simulations in Cadence were presented. More insight in the electrical characterization by considering experimental measurements with the integrated building blocks, is presented next.

A. MIXED-MODE WEIGHTING MULTIPLIER

The proposed analog-digital multiplier circuit, shown in Figure 2, is based on a highly linear programmable gain amplifier (PGA) and a sign circuit (SC). The PGA is based on the inherently linear current division principle when connecting two MOS transistor arrays under the same bias conditions [18]. It presents a simple digital control of the overall gain of the amplifier, thus emulating the weight multiplication in a neural processor.

The multiplier consists of a resistance R_1, an operational amplifier, and two arrays of PMOS transistors, M_1 and M_2, operating in a triode region and, thus, acting as active resistors. R_1 converts the input voltage V_{in} to a current, and M_1-M_2, set up the gain, weighting V_{in}. The transfer function

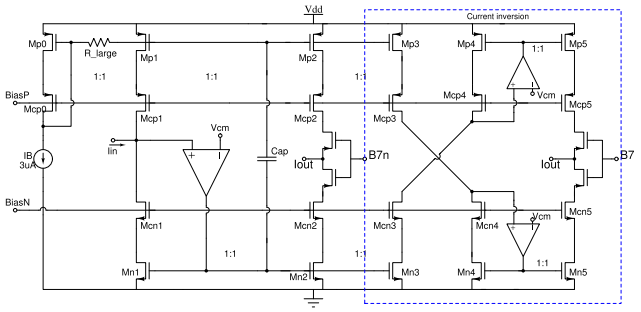

FIGURE 3. Sign circuit (SC) electronic topology.

of the proposed multiplier is defined by:

$$I_2 = \frac{1}{R_1}\frac{(W/L)_2}{(W/L)_1}V_{in} \tag{1}$$

The signal processing is carried out in the current domain and the activation coefficients (weights) are set by adjusting the equivalent size ratio (W/L) between M_1 and M_2 just turning on or off some of the transistors in the arrays. A maximum sizing ratio of "1" between the transistors is established, so that programmability is provided only by the transistor array M_2, whereas every transistor in parallel forming M_1 are always *on*.

Each multiplier is designed so that, the input is modulated by the 8-bit fixed point weight value stored in a digital register, which range from -127 to 127 [19]. Gain is controlled by the 7 least significant bits (LSB) in the digital word $(B_6 \ldots B_0)$, having 128 possible weight values, while the MSB (B_7) controls the operation sign, providing weight values from -1 to $+1$.

The sign circuit at the output of the multiplier determines the direction of the current, thus allowing for negative weights. The circuit was carefully designed and positioned in order to keep the PGA high linearity, just by setting a virtual ground at the M_2 source terminal (which is the same for the source voltage of M_1). Figure 3 shows the electronic topology. It is implemented with a dynamic class AB current mirror, providing two output branches: one for the forward current and the other for the inverted one. The topology is based on the quasi-floating gate approach to achieve class-AB operation [20], thus handling current levels higher than the bias current. By activating or deactivating the output branches with the MSB of the digital word, it is possible to select the current direction.

Finally, to carry out the addition of all the weighted signals, before being driven to the nonlinear circuit, a current approach was adopted for its simplicity, thus saving power and area consumption. The output of each multiplier is connected at the input of the transimpedance amplifier (TIA), thus providing a low impedance node for the addition of current signals, and also it carries out the current–voltage conversion. The transimpedance amplifier consists of a two-stage amplifier ($OA2$) and a feedback resistor R_2 (red dashed square in Figure 2). The schematic diagram of a neuron with

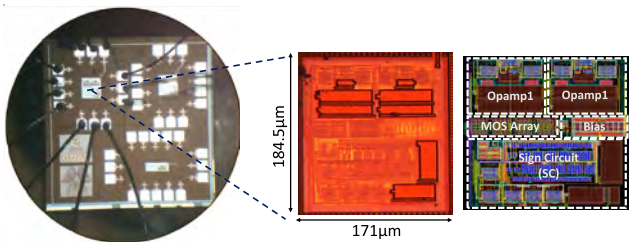

FIGURE 4. Microphotograph of the integrated 8-bit multiplier.

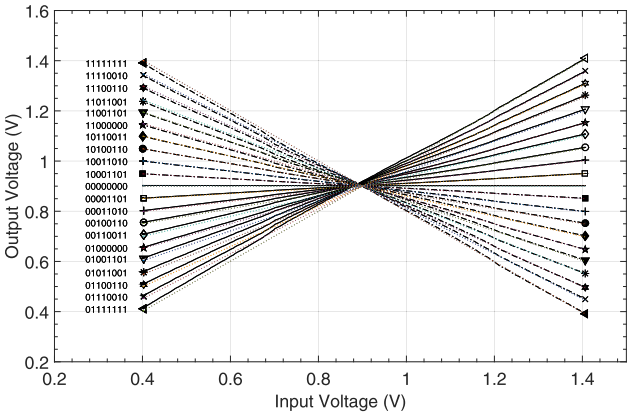

FIGURE 5. Integrated 8-bit multiplier characteristic considering ten digital words.

two weighted-signals combined is shown in Figure 2, and its output voltage is defined by:

$$V_{out} = \frac{R_2}{R_1}\cdot\left[\frac{(W/L)_2}{(W/L)_1}V_{in} + \frac{(W/L)_{2'}}{(W/L)_{1'}}V_{bias}\right] \tag{2}$$

Note that the proper addition of currents at the TIA input node is determined by the coupling between both the equivalent TIA input resistance (R_{TIA}) and the equivalent output resistance of all the multipliers connected in that node (R_{eq}). The first one is defined by $R_{TIA} = R_2/(1 + G_{OA2} \approx 5\Omega)$, whereas $R_{eq} = R_{SC}/m$ depends on the number of multipliers m connected in parallel and the sign circuit output impedance which is $R_{SC} \approx 30M\Omega$. Therefore, as long as $R_{TIA} \ll R_{eq}$, the coupling and thus the proper currents addition will be appropriate.

1) MULTIPLIER EXPERIMENTAL CHARACTERIZATION

Figure 4 shows the mixed-mode multiplier integrated prototype. The circuit is defined by an area of $171\mu m \times 184.5\mu m$ and its power consumption is 40μW by setting a maximum input current of $10\mu A$ through the input resistance R_1. In this sense, when the maximum gain is set by the digital word, which corresponds to $w = 1$, this current flows through M_1 and M_2 transistor arrays.

The output characteristic for ten digital words (with positive and negative values) is shown in Figure 5 by considering an input voltage range from $0.4V$ to $1.4V$. The ideal response is also shown in the figure with dotted lines. The integrated

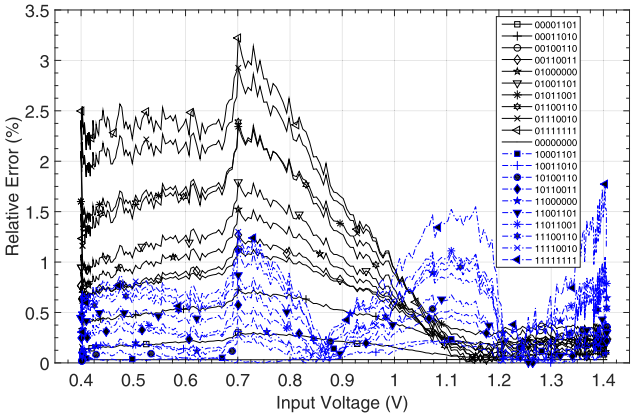

FIGURE 6. Relative error e_r of the integrated multiplier response when comparing with an ideal characteristic. Blue markers correspond to negative weights, while the black color for positive weights.

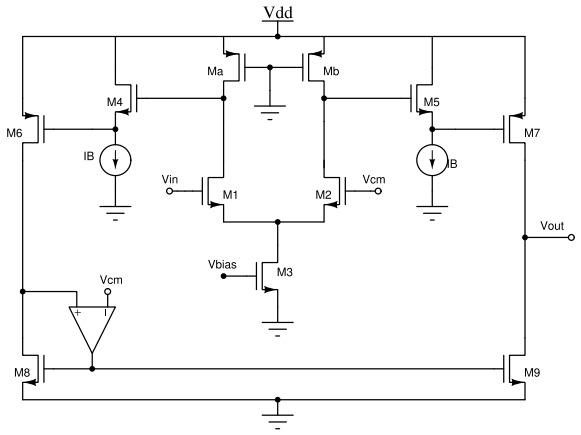

FIGURE 7. Non-linear activation function circuit.

circuit response shows a voltage offset of $25mV$, thus shifting all the curves to the left side. The circuit response is compared with the ideal one in order to verify its accuracy, and the relative error is presented in Figure 6. For positive weights, the maximum relative error is 3.25%, whereas for negative weights, the error decreases to 1.82%. In both cases, the maximum relative error occurs at the maximum gain setting.

B. NON-LINEAR ACTIVATION FUNCTION CIRCUIT

Several CMOS implementations of non-linear activation functions can be found in the literature. Most of them are designed to generate a response similar to a sigmoid or a *tanh* function. However, most of the designs lack of symmetry and the saturation levels are not well defined [21], [22], whereas other circuits show a pretty complex implementations with large area and high power consumption [23]. It is worth mentioning that despite the potential of the ReLU non-linear function in ML based implementations, this function is really efficient for deep learning neural architectures where layers with a high number of processors are required, and where the overall performance will not be affected if some neurons turn-off due to the well known dying ReLU problem.

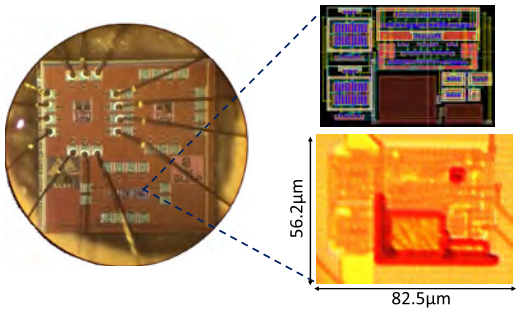

FIGURE 8. Integrated activation function circuit.

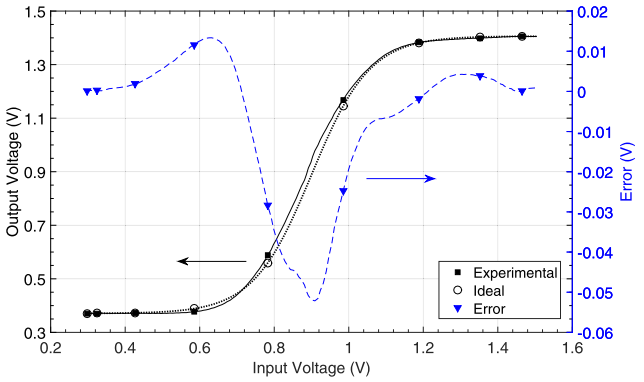

FIGURE 9. Integrated activation function circuit electrical characteristics.

The proposed activation function is based on the differential pair, because of the symmetrical output characteristic, and the well-defined maximum and minimum saturation levels. It consists of a differential NMOS input pair with active loads, particularly the transistors M_a and M_b operating in the linear region and is shown in Figure 7. The transistors M_4-M_5 act as level shifters, and the differential output is converted into a single output by current mirrors. The amplifier in the high-precision current mirror ($M_8 - M_9$) sets the common mode $V_{cm} = 0.9V$ at the output circuit, and is implemented by a simple differential pair with $40dB$ gain.

1) NON-LINEAR CIRCUIT EXPERIMENTAL CHARACTERIZATION

The integrated activation function circuit microphotograph is shown in Figure 8. The integration area is $82.5\mu m \times 56.2\mu m$ and it shows a power consumption less than $30\mu W$.

In order to compare the output characteristic with an ideal characteristic, a numerical hyperbolic tangent function was generated in MATLAB with the same gain and the same saturation levels. The response of the integrated circuit and the ideal function are shown in Figure 9, where a maximum error of $0.05V$ is appreciated when comparing both functions. The lower saturation level is about $0.4V$, whereas the upper saturation level corresponds to $1.4V$, and it shows the expected non-linear and symmetrical characteristic. The maximum gain of the integrated prototype is $G_{AF} = 3.3$ and Figure 10 shows the gain characteristic of both the prototype

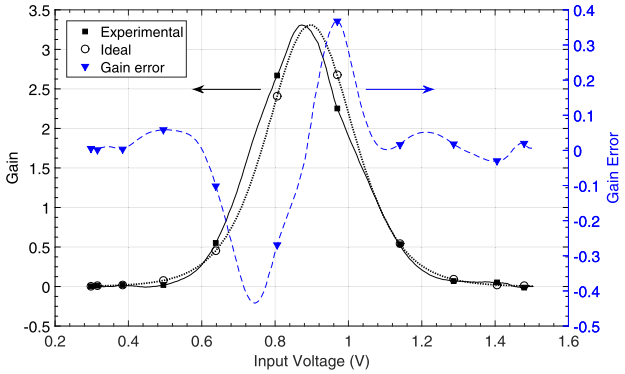

FIGURE 10. Integrated activation function circuit gain.

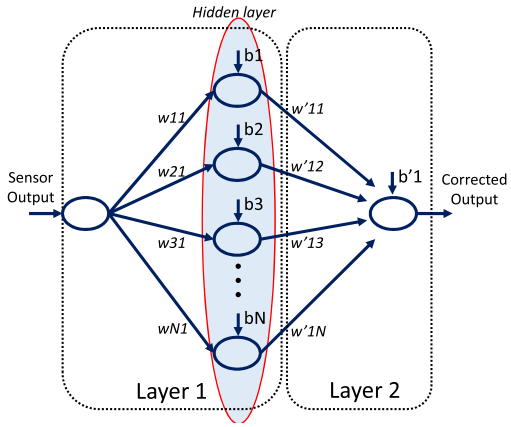

FIGURE 11. ML-based block diagram for sensor signal conditioning.

and the ideal function, as well as the error in gain. The maximum error in gain is $e_g = 0.4$, and it is as expected at the ends of the sigmoid shape, just before the saturation regions.

III. SELECTION OF MACHINE LEARNING ARCHITECTURE FOR SENSOR CONDITIONING

From the different machine learning based architectures designed for function approximation, multilayer perceptron (MLP) features make it a worthy candidate for use in sensor signal processing. The reduced set of arithmetic operations performed by these processors make them suitable to be implemented in small application-specific circuits [23]. A typical MLP diagram based on this architecture is shown in Figure 11. In our case, the system consists of two processing layers: the first one is integrated by the input node and the hidden layer with N neurons, and the second by the output layer. In the first layer, neurons weight the input data using the coefficients $wX1$, where X ranges from 1 to N. An additional signal called bias, with a constant value set to 1, increases the degrees of freedom of the system as it is multiplied by an extra variable coefficient, bX. The sum of both the weighted input and bias signals is also carried out in this layer, and the operation result is processed by a nonlinear operation in order to obtain the output of each processor in the hidden layer. A weighting of the neuron outputs from the hidden layer is carried out in the last layer, and an addition of the resulting signals with another weighted bias signal is made before providing the final system output [17].

Note that MLP architectures with more than one hidden layer could also be used, however, because of a trade-off between circuit complexity, power consumption and overall performance, a single hidden layer is enough to successfully address this kind of problem. If more layers were required, additional care must be taken in the electronic design in order to have a proper coupling between the resulting stages.

A. MLP HIGH-LEVEL MODELING FOR THE LEARNING PHASE

Once all the system components have been designed and individually characterized, it is required to verify their operation within a complete processing architecture. However, training the network to match the correct input-output transfer function is previously required. Because of the complexity neural structure and its high number of building blocks, the simulation of this tuning phase with the same microelectronic software requires programming in the corresponding high level tool, leading to long simulation times and not guaranteeing the convergence of the whole process.

As described in [17], to speed up the simulation in the training phase, a neural network toolbox provided by MATLAB is used to define the network architecture by incorporating the characteristics of the electronic system. In order to accomplish this, the main changes applied in the Neural Network Toolbox configuration are summarized next:

- Mathematical models that represent the operation of the electronic neurons need to be added to the toolbox, so that, high-level models of the arithmetics involved in the network were defined from the electrical characterization of the proposed building blocks.
- Bias definitions were discarded and an additional weighted input to the network at a constant voltage value was tied to all the processors.
- It was required to modify the network architecture by assigning a neural layer to each processor, and reconnecting the layers (now individual processors), thus emulating more realistic behaviors of the whole system.
- A second order learning algorithm based on the classical Levenberg-Marquardt error backpropagation was selected to speed up the weight fitting process. Although these algorithms present more mathematical complexity per epoch, the number of iterations are considerably reduced, thus providing a suitable performance at a reduced training time [24]. The algorithm was modified to limit the weight values to ± 1 range and, besides, an 8-bit discretization was carried out at each learning iteration.

IV. INTEGRATED MLP ELECTRICAL CHARACTERIZATION

Small perceptron network architectures (Fig. 11) are capable to compensate several types of sensor non-idealities by

TABLE 1. Digital weights obtained after the learning phase.

Weight (w)	Value	Digital Word	Weight (w)	Value	Digital Word	Weight (w)	Value	Digital Word
w_{11}	-1.0	11111111	w_{21}	0.9658	01111011	b_{11}	-0.5276	11000011
w_{12}	-0.2677	10100010	w_{22}	-0.2205	10011100	b_{12}	0.2283	00011101
w_{13}	0.2283	00011101	w_{23}	0.9449	01111000	b_{13}	0.8740	01101111
w_{14}	0.8583	01101101	w_{24}	0.5748	01001001	b_{14}	0.7323	01011101
w_{15}	-0.8031	11100110	w_{25}	0.3543	00101101	b_{15}	0.3701	00101111
						b_{26}	-1.0	11111111

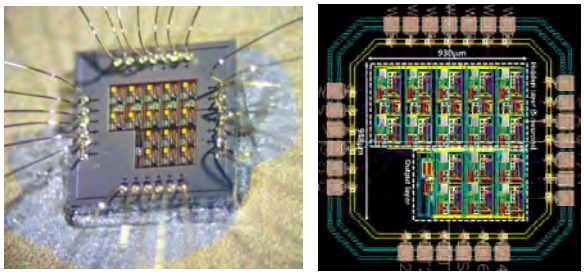

FIGURE 12. Integrated MLP prototype with 5 neurons in the hidden layer.

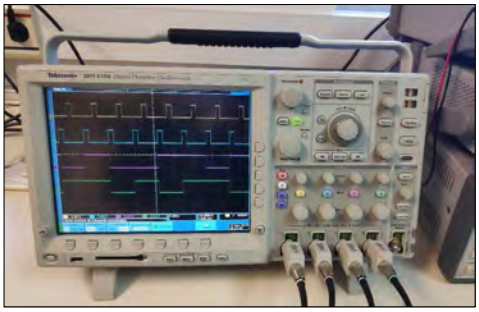

FIGURE 13. Clock and weight signals generated with the microcontroller.

correcting offset, gain and non-linearity errors. Particularly, an 1-5-1 architecture shows an outstanding trade-off between complexity, power consumption and correction capability, as it was demonstrated in the preliminary results presented in [17] at electronic simulation level. The integrated prototype electrical characterization is presented next in order to verify the proposed neural system proper performance by considering a sensor non-linear characteristic.

The application example is based on a thermistor non-linear response placed in a resistive divider. This sensor is considered because it is one of the most widely used low-cost devices for temperature measuring, but with a high non-linear degree. Its characteristic was used to train and simulate the proposed network. Note that the target or expected response corresponds with a linear shape with different gain and offset.

Figure 12 shows a microphotograph of the 1-5-1 MLP integrated in standard 0.18μm CMOS technology with $1.8V$ power supply. The layout of the circuit is also appreciated in the Figure, where it is possible to distinguish the two layers conforming the network system. It has an area of 0.86mm^2 and its power consumption is 1.4mW by considering a maximum input current of 10μA in each neuron.

8-bit registers were added in the integrated prototype in order to store and load the weights in each of the multipliers. In this way, the number of required pads and the integration area were reduced. In order to load the 8 bits defining each of the weights in a daisy chain, shift registers were implemented with cascade connected positive-edge triggered flip-flops (D flip-flops) by using a two-phase nonoverlapping, so that the hold time problems were avoided.

120 input–output patterns randomly selected were considered to carry out the learning phase. 70% of the dataset is used for training, 15% for test, and the remaining 15% for validation purposes. Once the MLP high-level model was trained, the system achieved the solution after 70 iterations and 16 8-bit digital weights were defined in order to weight the input and intermediate signals within the network architecture (5 for the input signal in the first layer, 6 for the bias signals and 5 for the first layer output signals). Table 1 summarizes the value of each one with its corresponding digital word.

The 16 weights were loaded serially, in each multiplier, by using an automatic digital control programmed in a microcontroller. An Arduino microcontroller was used to generate the complementary clock signals and each of the digital weights. The clock signals synchronization and two 8-bit encoded weights are shown in Figure 13. It is worth mentioning that a Mega/2560 board was used due to the number of available digital ports and for the EEPROM memory capability, which is required to store the obtained 8-bit weights.

The complete system requires a bias current $I_B = 500$nA, which is generated from a Keithley-2636B source meter. The bias signals of both the input and output layers were tied to a maximum voltage of $1.6V$, which later are weighted using the neural correction coefficients. The input signal was generated in order to emulate the output voltage of a thermistor placed in a resistive divider by considering a temperature range of $130°C$, so 20 input voltage values were considered along the whole non-linear characteristic. On other hand, the feedback resistance of the output transimpedance-amplifier (R_2 in Figure 2) was placed externally in order to adjust its value manually and thus ensure that it is capable of handling the appropriate current levels, if it was required. The test board

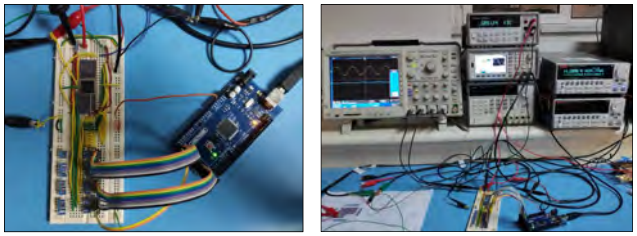

FIGURE 14. Photograph of the test board used for the characterization.

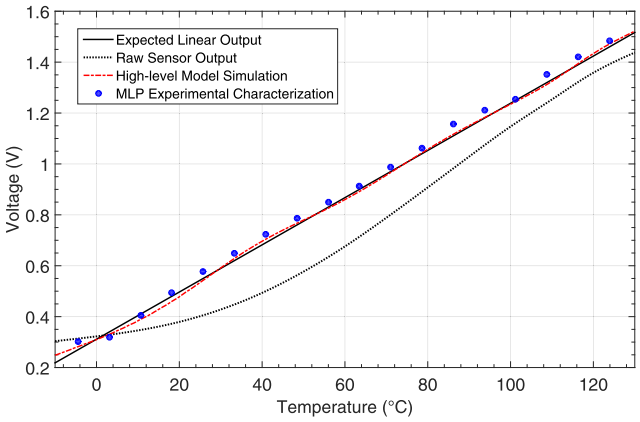

FIGURE 15. Thermistor non-linear output characteristic compared with the output obtained after the learning phase. Dashed and red line corresponds to the numerical high-level Matlab simulation, whereas the blue markers refers to the values obtained after experimental characterization of the MLP-based system.

TABLE 2. Comparison between the high-level simulation and experimental characterization.

Output Characteristic	Temp. Error e_T (°C)		Relative Error e_r (%)	
	Max.	Mean	Max.	Mean
Uncorrected*	21.35	12.64	27	15.73
Simulation*	2.84	0.97	4.90	1.82
Characterization*	3.89	1.80	5.0	2.7

* Temperature span between 0°C and 130°C.

used for the circuit characterization is shown in Figure 14, where a test setup used to load the neural weights and measure the output voltage is also presented.

Figure 15 shows the ML-based conditioning electronics output response obtained from experimental characterization considering 20 voltage values along the previously defined input range. Note that the target response (black and continuous line) and the characteristic obtained by high-level simulation in Matlab (red and dashed line) is also shown in order to compare them.

Figure 16 presents both the relative error (e_r) and the error in the temperature estimation (e_T) by considering the MLP electrical characterization. Results show a maximum relative error of 5%, which is almost 6 times less than the uncorrected response error within a temperature span of 130°C. Note that this error remains below 3% in most of this range.

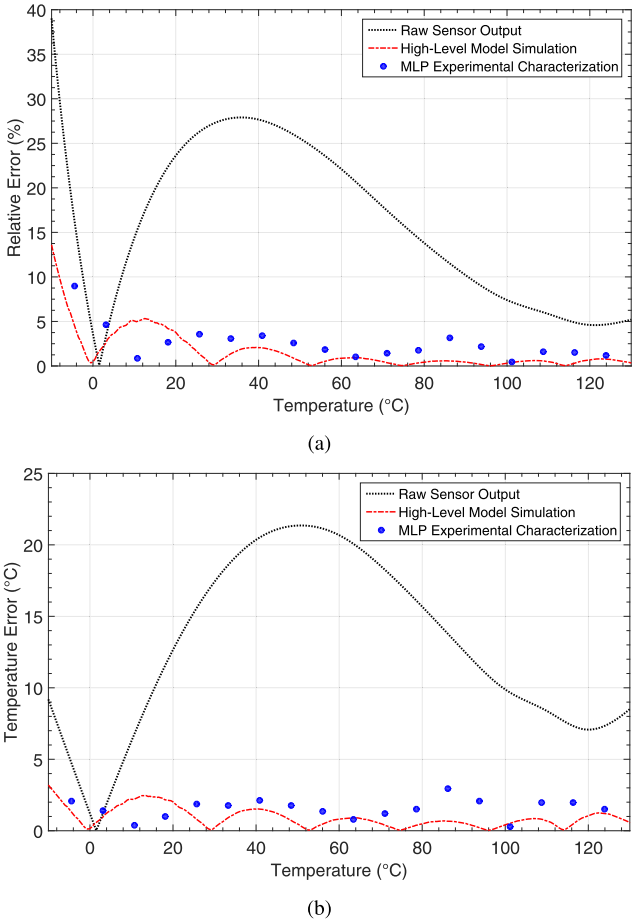

FIGURE 16. Comparison between the thermistor raw-output and the corrected output electrical characteristic: (a) relative error e_r and (b) error in temperature estimation e_T.

For the error in temperature estimation, the maximum value obtained is 3.9°C with a mean value of 1.8°C for the measured characteristic, which is a remarkable difference with the uncorrected characteristic, where a maximum error of 21.35°C is appreciated. These values show congruence with the high-level numerical simulation where a maximum error of 2.84°C is calculated.

Table 2 summarizes the calculated errors when comparing both the Matlab high-level simulation and measured responses, with the non-linear uncorrected thermistor characteristic. It is worth mentioning that if a chip-on-the-loop training approach had been adopted, a more accurate fit in the output network response would have been achieved. However, results presented above show congruence with the characteristic obtained from numerical model high-level simulations.

A. COMPARISON WITH OTHER IMPLEMENTATIONS

Finally, a comparative study of different adaptive based sensor conditioning circuits reported in literature is presented in Table 3. Most just linearize the sensor response or extend the linear section of the output characteristic. This is the main

TABLE 3. Comparison of different ML-based adaptive systems for sensor calibration.

Implementation	Sensor	Architecture	Algorithm	Dataset	Epochs	Power [mW]	Area [mm^2]	Remarks
Sarkar'13 [2] [Fully Analog]	Thermistor	Inverting amplifier	—	—	—	—	—	- Linearity approximately remains below 2% over a temperature range of 90°C. - The technique lacks of flexibility if different sensors are used.
Kumar'15 [12] [Digital]	Thermistor	VCO + MLP 1-3-1	Back-propagation	—	194	—	—	- 2-stage linearizing technique. - Good linearity is achived over a temperature range of 0°C - 100°C.
Rana'15 [13] [Digital]	Thermistor	MLP 1-2-2-1	Back-propagation	- 99 data	20000	—	—	- Temperature error below ±1°C in a range between 5°C and 65°C.
Zatorre'10 [14] [Mixed-Mode]	GMR	MLP 1-4-1	Weight Perturbation	- 175 patterns *90% for learning *10% for test	400	10.8	0.25	- 72% extension of the linear section where the error is less than 2°.
This work [Mixed-Mode]	Thermistor	MLP 1-5-1	Back-propagation	- 120 patterns *70% for training *15% for test *15% for validation	70	1.4	0.86	- Temperature mean error remains below 2° for most of the span. - Offset correction - Gain correction

difference with the proposed system, which in addition to linearizing, it also corrects offset and gain errors.

Digital implementations show a good linearity over a narrow temperature range, however, power consumption or complexity data are not specified [12], [13]. Note that a software solution requires memory cells and sequential machines, thus resulting in a larger occupied area and higher power consumption. Particularly, a non-linear digital function with a sigmoid characteristic can be generated in a microcontroller by means of approximations, multipliers and register operations; or by using Digital Signal Processor (DSP) slices, LUTs and shift registers when a Field Programmable Gate Array (FPGA) is considered. In either case, the resulting function is very similar to the ideal one, but requiring high computational resources and power consumption. The proposed analog implementation, on the other hand, requires less transistors for its generation, and the accuracy of the resulting activation function is in the same order as for the digital case. Moreover, digital ML based systems require higher resolution converters (ADCs) and the required resolution depends on the particular sensor non-linearity, thus reducing flexibility to the system in case a different sensor is required.

On other hand, a mixed-mode system [14] is also presented in Table 3. Although a different sensor is considered, it is possible to compare most of the electrical characteristics. Note that it shows a reduced area but a power consumption almost 10 times higher than the proposed system, thus making it unaffordable for integrated low-cost embedded systems.

Moderate linearity is appreciated in the fully-analog implementation [2]. It is based on a simple analog circuit, but it lacks of flexibility if different non-linear sensor is used.

V. CONCLUSION

A mixed-mode machine learning based conditioning circuit integrated in standard 0.18μm CMOS technology has been presented in this paper. The proposed system, in addition to linearize the sensor response, is capable to correct offset and gain errors.

The building blocks conforming the proposed adaptive system were designed and integrated in the same CMOS technology, and the high-level models obtained from their experimental characterization were used to verify the proper electronic behavior within a defined MLP architecture.

The integrated MLP 1-5-1 mixed-mode prototype was experimentally characterized by considering a thermistor non-linear response, and a comparison with similar implementations is also shown. Results showed that a maximum relative error of 27% before correction was reduced to less than 3% after compensation by considering an input span of 130°C, and a temperature estimation error below 3°C was achieved for the same temperature range.

It was demonstrated that the proposed system can correct the sensor response, considering offset, gain and non-linearity errors, with a reduced number of processors and power consumption less than 1.4mW.

Digital programmability provides flexibility to also compensate for deviations in system performance due to sensor aging. Furthermore, it allows the different types of sensors available in a measurement system to be calibrated according to their behavior with a single electronic architecture, as previously indicated in [17]. In this way, by storing in a small integrated memory the sets of weights obtained in the different trainings of the network, it is possible to alternatively compensate the behavior of several sensors using a single functional module and setting the appropriate set of weights for the correction required in every moment.

It is worth mentioning that by multiplexing the sensor inputs, the same circuitry will compensate several different sensors on the same sensing platform at a reduced size, cost and power consumption. In this sense, an universal sensor calibrator could be implemented.

Another additional advantage of the proposed adaptive system is that besides correcting the non-linear response of the sensor, it also compensates for all the implemented circuitry imperfections, thus obtaining a more robust system to process and parameter variations.

Finally, an interesting future research line is considering the possibility of adapting the network architecture depending on the problem complexity. This could be done by activating or deactivating individual processors within the system, depending on the non-linearity to be corrected, thus achieving a re-configurable adaptive system. In this way, power consumption could be reduced if required.

LIST OF ACRONYMS

The following abbreviations are used in the manuscript:

ADC	Analog-Digital Converter
AF	Activation Function
CMOS	Complementary Metal-Oxide-Semiconductor
DSP	Digital Signal Processor
FPGA	Field Programmable Gate Array
LUT	Look-Up-Table
LSB	Least Significant Bit
ML	Machine Learning
MLP	Multilayer Perceptron
MSB	Most Significant Bit
PGA	Programmable Gain Amplifier
PWL	Piecewise-Linear
ReLU	Rectified Linear Unit
SC	Sign Circuit
TIA	Transimpedance Amplifier

REFERENCES

[1] S. Bandyopadhyay, A. Das, A. Mukherjee, D. Dey, B. Bhattacharyya, and S. Munshi, "A linearization scheme for thermistor-based sensing in biomedical studies," *IEEE Sensors J.*, vol. 16, no. 3, pp. 603–609, Feb. 2016.

[2] A. R. Sarkar, D. Dey, and S. Munshi, "Linearization of NTC thermistor characteristic using op-amp based inverting amplifier," *IEEE Sensors J.*, vol. 13, no. 12, pp. 4621–4626, Dec. 2013.

[3] B. Murmann, "Digitally assisted analog circuits," *IEEE Micro*, vol. 26, no. 2, pp. 38–47, Mar. 2006.

[4] A. Flammini, D. Marioli, and A. Taroni, "Transducer output signal processing using an optimal look-up table in microcontroller-based systems," *Electron. Lett.*, vol. 33, no. 14, pp. 1197–1198, Jul. 1997.

[5] D. Sonowal and M. Bhuyan, "Linearizing thermistor characteristics by piecewise linear interpolation in real time FPGA," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Aug. 2013, pp. 1976–1980.

[6] G. van der Horn and J. Huijsing, *Integr. Smart Sensors: Design Calibration* (The Springer International Series in Engineering and Computer Science). Cham, Switzerland: Springer, 1997.

[7] G. Bucci, M. Faccio, and C. Landi, "New ADC with piecewise linear characteristic: Case study-implementation of a smart humidity sensor," *IEEE Trans. Instrum. Meas.*, vol. 49, no. 6, pp. 1154–1166, Dec. 2000.

[8] A. J. López-Martin, M. Zuza, and A. Carlosena, "A CMOS A/D converter with piecewise linear characteristic and its application to sensor linearization," *Analog Integr. Circuits Signal Process.*, vol. 36, nos. 1–2, pp. 39–46, Jul. 2003.

[9] J. M. D. Pereira, O. Postolache, and P. M. B. S. Girao, "A Digitally Programmable A/D Converter for Smart Sensors Applications," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 1, pp. 158–163, Feb. 2007.

[10] E. Alpaydin, *Introduction to Machine Learning*, 2nd ed. Cambridge, MA, USA: MIT Press, 2010.

[11] T. Islam, S. Ghosh, and H. Saha, "ANN-based signal conditioning and its hardware implementation of a nanostructured porous silicon relative humidity sensor," *Sens. Actuators B, Chem.*, vol. 120, no. 1, pp. 130–141, Dec. 2006.

[12] V. N. Kumar, K. V. L. Narayana, A. Bhujangarao, and S. Sankar, "Development of an ANN-based linearization technique for the VCO thermistor circuit," *IEEE Sensors J.*, vol. 15, no. 2, pp. 886–894, Feb. 2015.

[13] K. P. S. Rana, N. Mittra, N. Pramanik, P. Dwivedi, and P. Mahajan, "A virtual instrumentation approach to neural network-based thermistor linearization on field programmable gate array," *Experim. Techn.*, vol. 39, no. 2, pp. 23–30, Mar. 2015.

[14] G. Zatorre, N. Medrano, M. T. Sanz, B. Calvo, P. A. Martinez, and S. Celma, "Designing adaptive conditioning electronics for smart sensing," *IEEE Sensors J.*, vol. 10, no. 4, pp. 831–838, Apr. 2010.

[15] B. Zamanlooy and M. Mirhassani, "An analog CVNS-based sigmoid neuron for precise neurochips," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 3, pp. 894–906, Mar. 2017.

[16] T. Talaska, M. Kolasa, R. Dlugosz, and W. Pedrycz, "Analog programmable distance calculation circuit for winner takes all neural network realized in the CMOS technology," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 3, pp. 661–673, Mar. 2016.

[17] J. A. Martínez-Nieto, N. Medrano-Marqués, M. T. Sanz-Pascual, and B. Calvo-López, "High-level modeling and simulation tool for sensor conditioning circuit based on artificial neural networks," *Sensors*, vol. 19, no. 8, p. 1814, Apr. 2019.

[18] M. T. Sanz, S. Celma, and B. Calvo, "Using MOS current dividers for linearization of programmable gain amplifiers," *Int. J. Circuit Theory Appl.*, vol. 36, no. 4, pp. 397–408, 2008.

[19] D. Anguita, S. Ridella, and S. Rovetta, "Worst case analysis of weight inaccuracy effects in multilayer perceptrons," *IEEE Trans. Neural Netw.*, vol. 10, no. 2, pp. 415–418, Mar. 1999.

[20] F. Esparza-Alfaro, A. J. Lopez-Martin, R. G. Carvajal, and J. Ramirez-Angulo, "Highly linear micropower class AB current mirrors using quasi-floating gate transistors," *Microelectron. J.*, vol. 45, no. 10, pp. 1261–1267, Oct. 2014.

[21] G. Khodabandehloo, M. Mirhassani, and M. Ahmadi, "Analog implementation of a novel resistive-type sigmoidal neuron," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 4, pp. 750–754, Apr. 2012.

[22] J. Shamsi, A. Amirsoleimani, S. Mirzakuchaki, A. Ahmade, S. Alirezaee, and M. Ahmadi, "Hyperbolic tangent passive resistive-type neuron," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 581–584.

[23] G. Zatorre-Navarro, N. Medrano-Marques, and S. Celma-Pueyo, "Analysis and simulation of a mixed-mode neuron architecture for sensor conditioning," *IEEE Trans. Neural Netw.*, vol. 17, no. 5, pp. 1332–1335, Sep. 2006.

[24] J. J. Moré, "The Levenberg-Marquardt algorithm: Implementation and theory," in *Numerical Analysis*, G. A. Watson, Ed. Berlin, Germany: Springer, 1978, pp. 105–116.

JAVIER A. MARTÍNEZ-NIETO was born in Mexico. He received the M.Sc. and Ph.D. degrees in electronic engineering from the National Institute for Astrophysics, Optics and Electronics (INAOE), Puebla, Mexico, in 2013 and 2018, respectively. He is currently a Postdoctoral Research Associate with the Electronic Design Group (GDE), University of Zaragoza, Spain. His current research interests include analog and mixed-mode CMOS IC design, signal conditioning circuits, integrated sensor interfaces, and on-chip programmable circuits.

MARIA TERESA SANZ-PASCUAL (Member, IEEE) received the Ph.D. degree in electronic engineering from the University of Zaragoza, Spain, in 2004. She was a member of the Electronic Design Group, Department of Electronic Engineering and Communications, University of Zaragoza, until 2008. She is currently a Full Researcher with the Electronics Department, National Institute for Astrophysics, Optics and Electronics, Mexico. Her research interests include analog and mixed IC design, integrated sensor readouts, and power conversion for energy harvesting.

N. MEDRANO (Senior Member, IEEE) received the B.Sc. and Ph.D. degrees in physics from the University of Zaragoza, Spain, in 1989 and 1998, respectively. He is currently an Associate Professor of electronics with the Faculty of Physics, University of Zaragoza, and a member of the Group of Electronic Design, Aragon Institute of Engineering Research, University of Zaragoza. His current research interests include implementation of neural networks for signal processing, integrated sensor interfaces, wireless sensor networks, and intelligent instrumentation.

BELEN CALVO LOPEZ (Senior Member, IEEE) received the B.Sc. degree in physics and the Ph.D. degree in electronic engineering from the University of Zaragoza, Spain, in 1999 and 2004, respectively. She is a member of the Group of Electronic Design, Aragon Institute of Engineering Research (GDE-I3A), University of Zaragoza. Her research interests include analog and mixed-mode CMOS IC design, on-chip programmable circuits, integrated optical receivers, low-voltage low-power monolithic sensor interfaces, and wireless sensors networks.

DIEGO ANTOLÍN CAÑADA received the M.Sc. and Ph.D. degrees in electronic engineering from the University of Zaragoza, Spain, in 2012 and 2017, respectively. He is currently an Associate Professor of electronics with the Escuela Politécnica de la Almunia, University of Zaragoza, and a member of the Group of Electronic Design, Aragon Institute of Engineering Research, University of Zaragoza. His current research interests include power electronics, wireless sensor networks, the Internet of Things, and intelligent instrumentation.