E-ISSN: 2676-6167 | P-ISSN: 2538-5100

Journal of Applied Research on Industrial Engineering

www.journal-aprie.com

J. Appl. Res. Ind. Eng. Vol. 8, No. 3 (2021) 205–212.

Paper Type: SI: ADLRTCA

Designing Novel LDO Voltage Regulator Implementation on FPGA Using Neural Network

Mahdieh Jahangiri[1], Ali Farrokhi[2], * , Amir Amirabadi[3]

[1] Department of Electrical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran; mahdiehjahangiri@yahoo.com;
[2] Department of Electrical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran; ali_farrokhi@azad.ac.ir;
[3] Department of Electrical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran; amirabadi@gmail.com.

Citation:

Jahangiri, M., Farrokhi, A., & Amirabadi, A. (2021). Designing novel LDO voltage regulator implementation on FPGA using neural network. *Journal of applied research on industrial engineering, 8*(3), 205-212.

Received: 24/01/2021 Reviewed: 02/03/2021 Revised: 02/05/2021 Accept: 05/05/2021

Abstract

This paper describes a new technique for implementing an Artificial Neural Network (ANN) using Field Programmable Gate Array (FPGA). The goal is design the Low Drop Output (LDO) voltage-regulator circuit with the desired features depending on the application. (The first novelty is designing an LDO with variable features). Voltage regulators bring voltage changes to a stable and acceptable level, especially for products using portable devices. The fragmentary neural network algorithm is modeled using the Xilinx generator system and it can be implemented in Xilinx FPGA (the second novelty is implanting fragmentary ANN in FPGA for parallel computations and real time design). The neural network is trained using the levenberg-Marquardt algorithm which is the data collected from HSPICE software. In Matlab, the tangent-sigmoid function is used as a neuron activation function, but the block set provided by the Xilinx generator system does not have a tangent-sigmoid operator, so the tan-sigmoid operator is modeled on the Maclaurin expansion (the third novelty is using Maclaurin series for approximation function along with the reduction of connections in the neural network to reduce many blocks in FPGA). In this paper, the similarity of the tangent-sigmoid function produced using Matlab and the approximation of the performance of this function using the Maclaurin series are shown. When the inputs are between -0.5 to +0.5, the simulated results show that the absolute error between the values of tan-sigmoid function based on Matlab and Xilinx System Generator using Maclaurin power series are not more than 0.17%. The performance modeling of the system generator with 0.996515% accuracy of Matlab modeling.

Keywords: Neural network, System generator, LDO voltage regulator, Field programmable gate array, Multilayer perceptron.

1 | Introduction

Voltage regulators are widely used in integrated circuits. Voltage regulators bring voltage changes to a stable and acceptable level, especially for products using portable batteries such as laptops, mobiles, and cameras. Voltage regulators have some features that, depending on the type of application, have some advantages over others, such as power Consumption, transient time, Power Supply Rejection Ratio (PSRR), output voltage ripple, line adjustment and load adjustment, but the relationship

Corresponding Author: ali_farrokhi@azad.ac.ir

10.22105/JARIE.2021.270076.1244

between these features is nonlinear. Designers of analog circuits face many difficulties and challenges in realizing non-linear problems in [1] and [2]. They usually take a long time to try and find the best option. The difficulties of nonlinearity analogue circuit design and the ability of neural network to modeling, categorizing and predicting nonlinear systems with the help of parallel computing, made a large number of literature that use artificial neural network to design analogue circuit, but there have poor generalization ability [3] and [4]. Field Programmable Gate Arrays (FPGAs) due to features such as extensive parallelization, high retuning, high efficiency, and low power consumption are nowadays one of the best options for implementing artificial neural networks in [5]-[7].

In recent years, the use of evolutionary algorithms to optimize electronic circuits has increased. Electronic circuits are optimized with different features. In [8] reference conductors and in reference in [8]-[10]. LDO voltage regulators used evolutionary algorithms to design, but in all of them only a few blocks of the system such as error amplifiers were optimized in this way. There are also challenges; the search space should be large enough to include a large set of feasible and optimal solutions.

The novation in this paper over several experiments is using the macloran series to approximate the activation function of the sigmoid tangent in the hidden layer of the neural network. In fact, this approximation increases the speed and reduce the computational time of the neural network as well as reducing the FPGA computing blocks. The former attempts to avoid being trapped in the local optima and the later can rapidly find the accurate solution to accelerate its evolving speed.

In this paper, a sufficient sample space for the neural network was provided by using various simulations in Hspice software. FPGA implementation has been used to speed up operations to find the best solution. In the following Part 2, the LDO voltage regulator and specifications for the specific applications are presented. In Part 3, the details of the neural network used are discussed. Part 4 is about the implementation on the FPGA and comparison of the results will be concluded in Part 5.

2 | LDO Voltage Regulator

Today's battery-powered portable devices have a power management system. The task of this power management system is to provide a constant voltage ideal for high-frequency analog and high-voltage noise-sensing blocks [11]-[14]. Parameters such as small output capacitor, low power supply, low output voltage drop and low output ripple voltage are monitored for this task [15]-[19]. There are several parameters to consider when optimizing. In this paper, an LDO with a small output capacitor and a low output ripple is used for optimization for on-chip applications in [20]. The LDO circuit used in *Fig. 1* is illustrated. Using, Monte Carlo simulation in Hspice for non-linear properties of Wm11, Cb, Cout, IB elements including power consumption, transient time, ripple feed ratio and line adjustment were measured, and a good sample space was ready for use on the neural network in [21] and [22].

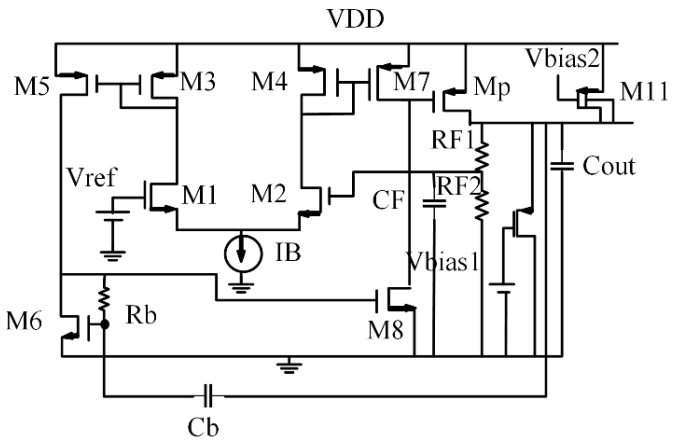


Fig. 1 . LDO circuit used for optimization in [17].

3 | Neural Network

There are several training algorithm. Lately, the Unscented Kalman Filter (UKF) in [23] has gained a lot of popularity with Extended Kalman Filter (EKF) because of it's the neural network is consisted of a large number of neurons, as processing elements that are connected in parallel in [24]. Each of the fittings has a weighting factor. These weights are adjusted during the training process. The most usual training algorithm is Levenberg-Marquardt in [25]. Many operations can usually be performed on a PC with an available solution. Whenever backpropagation in [26], EKF in [27], and [28] are used the solution required moderate.

Number of math operations and suitability for performing on hardware. A contrast between EKF and backpropagation is presented in [29], where EKF effectiveness in learning is higher due to its high convergence speed and this is presented in stability and performance to estimate nonlinear systems, but its computational burden is much greater than EKF. Sliding Mode Control Theory (SMCT) in [30]. SVSF is powerful for modeling uncertainties, ensures high convergence rate and ensures convergence of estimation error inside a border layer. In addition, this is an attractive way to train ANN because of its ability to converge Minimum number of epochs in [31]. For an estimation example of a nonlinear system, the reader can see [32]. LMA interpolates between the Gauss-Newton Algorithm (GNA) and the descending gradient method. It makes the LMA more robust than the GNA, which means; in many cases find an answer, even if it starts far beyond the final minimum. This method used to minimize the sum of the squares of error over a training period to find the minimum of a multivariable nonlinear function.

$$J(w) = 1/2 \sum_{s=1}^{p} \sum_{i=1}^{Nm} \left(d_{s,i} - y_{s,i}^{M} \right)^{2}. \quad (1)$$

In "*Eq. (1)*"the objective value (d) of i is the actual output (y) of the training pattern of (S), the weight vector (w) generated by all the weight coefficients and bias of the network, and the number of outputs. In this method, the initial weight coefficients vector is corrected according to "*Eq. (2)*"to reach the optimum value of the learning rate (μ) and is always positive, the input matrix, and the Jacobian error matrix (J) [33].

$$w_{k+1} = w_k - (J_K^T J_k + \mu I)^{-1} J_k e_K. \quad (2)$$

The artificial neural network implemented by Matlab software is shown in *Fig. 2*. After trying different architectures for the neural network in question, the neural network with a hidden layer with the tangent sigmoid activation function is selected. Three Hspice outputs including PSRR, transient time and ripple output voltage considered as three input neurons and outputs to be optimized by the neural network. Includes Wout, Cb, Cout, IB, so the numbers of neurons of the neural network output are four.

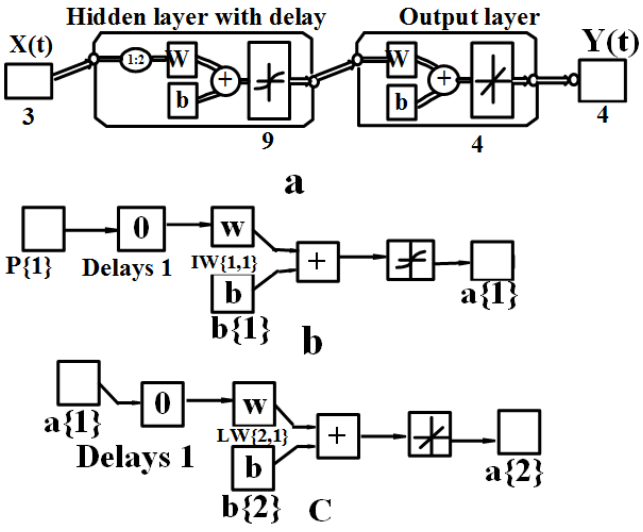


Fig. 2. a) neural network architecture, b) first layer architecture, c) second layer architecture.

After completing the training phase and determining the weights and bias values for the layers obtained in the next step, a neural network model is generated using Simulink Matlab as shown in *Fig. 2.a*, *Fig. 2.b* and *Fig. 2.c* illustrates the architecture of the first layer and the output of the neural network. The connection between the input layer and hidden layer is not fully connected meaning, instead of connecting each neuron of hidden layer to three neurons of the input layer, it only connects to two input layer neurons, also connections of output layer to hidden layer is not in complete form. Finally, connections are in a form that can use all of the input layer information. After the simulation, the number of using data and the output regression are shown in *Table 1*.

Table 1. The output regression.

Number of data	Training	Validation	Test
Regression	0.99766	0.99776	0.98498

4 | Implementation on FPGA and Compare Results.

Xilinx system generator is a high level, powerful tool for converting Matlab models to FPGA hardware. To implement a neural network with a system generator, some blocks such as a tangent sigmoid is not present in this tool, so for activation function approximation Macloran series is used. "*Eq. (3)*" represents the tangent function of sigmoid and "*Eq. (4)*"provides the Maclaurin expansion.

$$f(x)=\frac{(e^{x}-e^{-x})}{(e^{x}-e^{-x})}. \tag{3}$$

$$f(x)=f(0)+f'(0)x+\frac{f''(0)}{2!}x^{2}+\frac{f'''(0)}{3!}x^{3}. \tag{4}$$

In this case the approximation of the tangent sigmoid function using the Maclaurin series can be as "*Eq. (5)*".

$$f(x)=x-0.33x^{3}+0.133x^{5}. \tag{5}$$

Fig. 3 shows the real tangent sigmoid function and its polynomial approximation in range of -5 to +5.

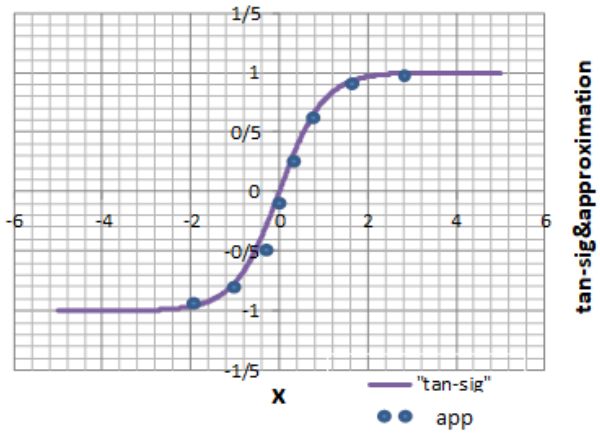


Fig. 3. Comparison between the real sigmoid tangent function and its approximation tables.

As mentioned above, optimization with macloran series and implementation on FPGA becomes the best choice to solve the designing problems so the basic idea is simple, using macloran series to reach the optimal approximation of activation functions, and then using BP algorithm to find the accurate value of each parameter. The detailed of proposed scheme is shown as *Fig. 4*.

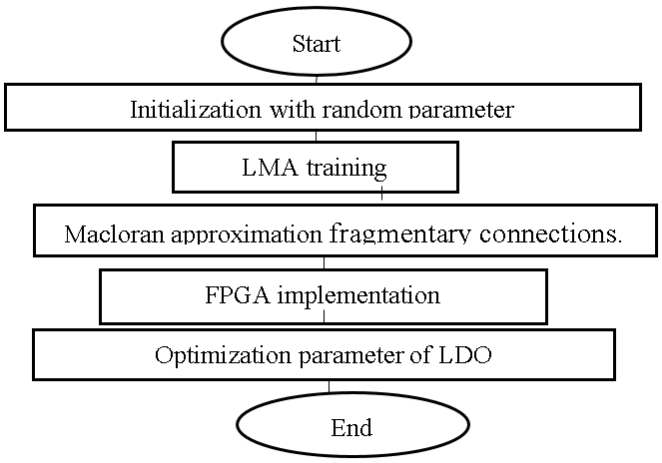
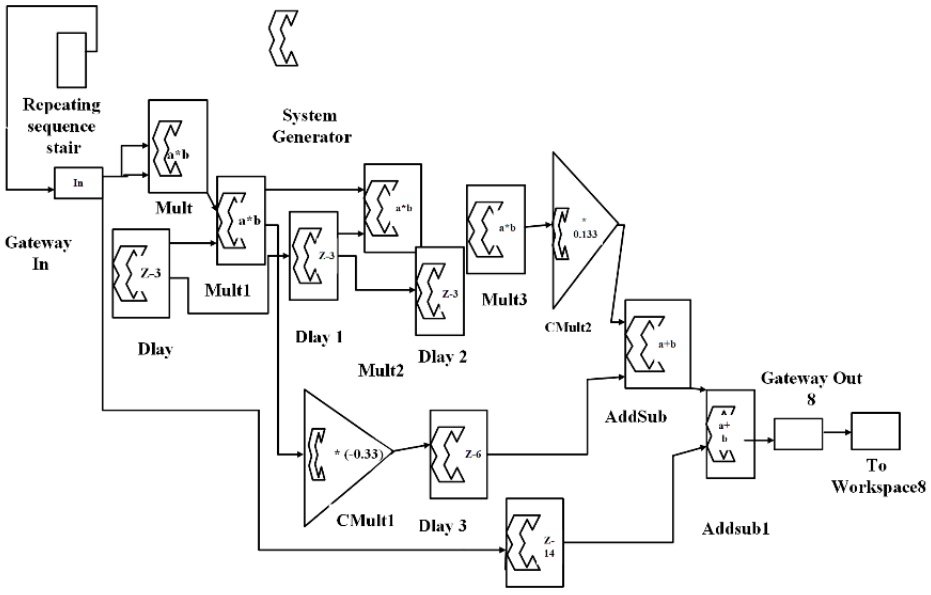


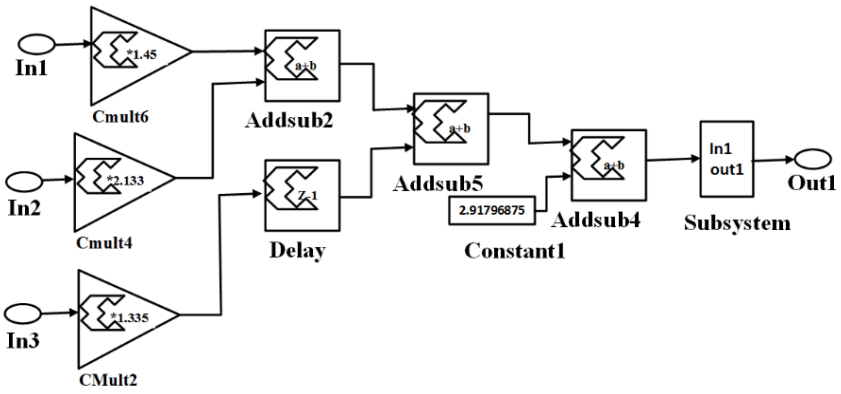
Fig. 4. Detail diagram of proposed method.

Fig. 5 illustrates the schemes related to the implementation of the entire neural network, single neuron, and activation function. Instead of using a neural network with full connections, a fragmentary neural network with broken connections is used. Advantages of using fragmentary neural network in FPGA is due to the fact that in FPGA each adder gate can only have two inputs, in addition to reducing number of gates at the surface and in depth, the speed of circuit and operation frequency increase without reducing the input information to the box. *Fig. 5.c* View of the neural network with fragmentary connections.

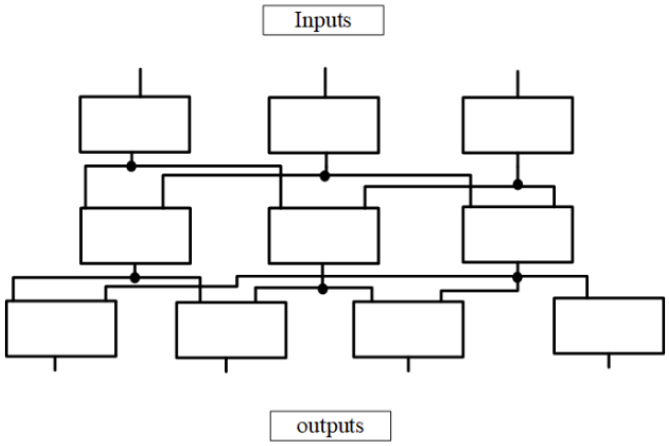
ISE software has been used for FPGA planning. ISE creates a bit file that can be uploaded to FPGA. The simulated results of the FPGA implementation of the tangent sigmoidal approximation function using the Maclaurin series Xilinx System Generator is shown in *Table 2*. The performance for the neural network to the model implemented on the generator system is shown in *Table 3*.



a



b



c

Fig. 5. a) activation function b) single neuron c) whole network.

Table 2. Simulation results of the sigmoid tangent function.

X	Tan-sig approximation	Theoretical tan-sig	Error %
+5	+1	+1	0
-0.5	-0.4629	-0.4621	0.17
-0.3	-0.2914	-0.2913	0.034
-0.1	-0.0997	-0.0997	0
0	0	0	0
0.2	0.1974	0.1974	0
0.4	0.3802	0.3799	0.078
0.5	0.4629	0.4621	0.17
-5	-1	-0.9908	0.0092

Table 3. Compares the performance for the neural network to the model implemented on the generator system.

Predicting Circuit Parameter Values	Neural Network in Simulink Matlab	Neural network in Xilinx Generator System with Approximation of Macloran
	0.99776	0.996515

5 | Conclusion

In this paper, the neural network method is used to design LDO voltage regulator with arbitrary features. The results show that the approximation used in the activation function along with the reduction of connections in the neural network are used to minimize the logical adjustments and improve the proposed neural network computation time. It is very useful to use top-level tools such as the system generator to validate and design any set of pattern recognition and time series forecasting algorithms. The innovation of the paper is multifaceted; one is that approximation is reduced as the volume of computation reaches the answer; second, this implementation on FPGA is a real-time design of a LDO circuit with the desired features, which is very significant in military power management applications.

References

[1] Torres, J., El-Nozahi, M., Amer, A., Gopalraju, S., Abdullah, R., Entesari, K., & Sanchez-Sinencio, E. (2014). Low drop-out voltage regulators: Capacitor-less architecture comparison. *IEEE circuits and systems magazine*, *14*(2), 6-26. DOI: 10.1109/MCAS.2014.2314263

[2] Mohamed, A. R., Qi, L., & Wang, G. (2021). A power-efficient and re-configurable analog artificial neural network classifier. *Microelectronics journal*, *111*, 105022. https://doi.org/10.1016/j.mejo.2021.105022

[3] Zhang, L., Zheng, C., Li, T., Xing, L., Zeng, H., Li, T., ... & Zhou, Z. (2017). Building up a robust risk mathematical platform to predict colorectal cancer. *Complexity*, *2017*. https://doi.org/10.1155/2017/8917258

[4] Sahani, M., & Dash, P. K. (2020). FPGA-based semisupervised multifusion RDCNN of process robust VMD data with online kernel RVFLN for power quality events recognition. *IEEE transactions on neural networks and learning systems*, 1-13. DOI: 10.1109/TNNLS.2020.3027984

[5] Ma, Y., Suda, N., Cao, Y., Seo, J. S., & Vrudhula, S. (2016, August). Scalable and modularized RTL compilation of convolutional neural networks onto FPGA. *2016 26th international conference on field programmable logic and applications (FPL)* (pp. 1-8). IEEE. DOI: 10.1109/FPL.2016.7577356

[6] Zhu, J., & Sutton, P. (2003, September). FPGA implementations of neural networks–a survey of a decade of progress. *International conference on field programmable logic and applications* (pp. 1062-1066). Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-540-45234-8_120

[7] Malathi, S., & Jayachandran, J. (2020). FPGA implementation of NN based LMS–LMF control algorithm in DSTATCOM for power quality improvement. *Control engineering practice*, *98*, 104378. https://doi.org/10.1016/j.conengprac.2020.104378

[8] Sanabria-Borbon, A. C., & Tlelo-Cuautle, E. (2014, August). Sizing analog integrated circuits by combining g m/I D technique and evolutionary algorithms. *2014 IEEE 57th international midwest symposium on circuits and systems (MWSCAS)* (pp. 234-237). IEEE. DOI: 10.1109/MWSCAS.2014.6908395

[9] Chinchore, J. B., & Thakker, R. A. (2015, March). Design of low dropout regulator using artificial bee colony evolutionary algorithm. *2015 international conference on circuits, power and computing technologies [ICCPCT-2015]* (pp. 1-8). IEEE. DOI: 10.1109/ICCPCT.2015.7159280

[10] Lopez-Arredondo, J., Tlelo-Cuautle, E., & Trejo-Guerra, R. (2015, March). Optimizing an LDO voltage regulator by evolutionary algorithms considering tolerances of the circuit elements. *2015 16th latin-american test symposium (LATS)* (pp. 1-5). IEEE. DOI: 10.1109/LATW.2015.7102506

[11] Ho, M., Leung, K. N., & Mak, K. L. (2010). A low-power fast-transient 90-nm low-dropout regulator with multiple small-gain stages. *IEEE journal of solid-state circuits*, *45*(11), 2466-2475. IEEE. DOI: 10.1109/JSSC.2010.2072611

[12] Tang, J., Lee, J., & Roh, J. (2018). Low-power fast-transient capacitor-less LDO regulator with high slew-rate class-AB amplifier. *IEEE transactions on circuits and systems II: express briefs*, *66*(3), 462-466. IEEE. DOI: 10.1109/TCSII.2018.2865254

[13] Lim, C. C., Lai, N. S., Tan, G. H., & Ramiah, H. (2015). A low-power fast transient output capacitor-free adaptively biased LDO based on slew rate enhancement for SoC applications. *Microelectronics journal*, *46*(8), 740-749. https://doi.org/10.1016/j.mejo.2015.06.002

[14] Qu, X., Zhou, Z. K., & Zhang, B. (2015). Ultralow-power fast-transient output-capacitor-less low-dropout regulator with advanced adaptive biasing circuit. *IET circuits, devices & systems*, *9*(3), 172-180. DOI: 10.1049/iet-cds.2014.0162

[15] Fathipour, R., Saberkari, A., Martinez, H., & Alarcón, E. (2014). High slew rate current mode transconductance error amplifier for low quiescent current output-capacitorless CMOS LDO regulator. *Integration*, *47*(2), 204-212. https://doi.org/10.1016/j.vlsi.2013.10.005

[16] Lee, Y. H., & Chen, K. H. (2010, August). A 65nm sub-1V multi-stage low-dropout (LDO) regulator design for SoC systems. *2010 53rd IEEE international midwest symposium on circuits and systems* (pp. 584-587). IEEE. DOI: 10.1109/MWSCAS.2010.5548893

[17] Milliken, R. J., Silva-Martínez, J., & Sánchez-Sinencio, E. (2007). Full on-chip CMOS low-dropout voltage regulator. *IEEE transactions on circuits and systems I: regular papers*, *54*(9), 1879-1890. IEEE. DOI: 10.1109/TCSI.2007.902615

[18] Guo, J., & Leung, K. N. (2010). A 6-$\mu $ W Chip-area-efficient output-capacitorless LDO in 90-nm CMOS technology. *IEEE journal of solid-state circuits*, *45*(9), 1896-1905. IEEE. DOI: 10.1109/JSSC.2010.2053859

[19] Saberkari, A., Alarcón, E., & Shokouhi, S. B. (2013). Fast transient current-steering CMOS LDO regulator based on current feedback amplifier. *Integration*, *46*(2), 165-171. https://doi.org/10.1016/j.vlsi.2012.02.001

[20] Jahangiri, M., & Farrokhi, A. (2019). Fast transient capacitor-less low-dropout regulator based on output voltage spike reduction circuits for SOC applications. *Journal of Circuits, Systems and Computers*, *28*(03), 1950043. https://doi.org/10.1142/S0218126619500439

[21] Mathews, V. J. (1991). Adaptive polynomial filters. *Jul*, *1*, *8*(3) 10-26. IEEE. DOI: 10.1109/79.127998

[22] Wen, H., Li, T., Chen, D., Yang, J., & Che, Y. (2021). An optimized neural network classification method based on kernel holistic learning and division. *Mathematical problems in engineering*, *2021*. https://doi.org/10.1155/2021/8857818

[23] Rigatos, G. G. (2011). A derivative-free Kalman filtering approach to state estimation-based control of nonlinear systems. *IEEE transactions on industrial electronics*, *59*(10), 3987-3997. IEEE. DOI: 10.1109/TIE.2011.2159954

[24] Kshirsagar, P., & Rathod, N. (2012). Artificial neural network. *International journal of computer applications*, *2*, 12-16.

[25] Reynaldi, A., Lukas, S., & Margaretha, H. (2012, November). Backpropagation and Levenberg-Marquardt algorithm for training finite element neural network. *2012 sixth UKSim/AMSS european symposium on computer modeling and simulation* (pp. 89-94). IEEE. DOI: 10.1109/EMS.2012.56

[26] Wang, S., Ling, X., Zhang, F., & Tong, J. (2010, March). Speech emotion recognition based on principal component analysis and back propagation neural network. *2010 international conference on measuring technology and mechatronics automation* (Vol. 3, pp. 437-440). IEEE.

[27] Haykin, S. S. (2008). *Adaptive filter theory*. Pearson Education India.

[28] Mercorelli, P. (2016). A motion-sensorless control for intake valves in combustion engines. *IEEE transactions on industrial electronics*, *64*(4), 3402-3412. IEEE. DOI: 10.1109/TIE.2016.2598314

[29] Li, S. (2001, July). Comparative analysis of backpropagation and extended Kalman filter in pattern and batch forms for training neural networks. *IJCNN'01. international joint conference on neural networks. proceedings (Cat. No. 01CH37222)* (Vol. 1, pp. 144-149). IEEE. DOI: 10.1109/IJCNN.2001.939007

[30] Young, K. D., Utkin, V. I., & Ozguner, U. (1999). A control engineer's guide to sliding mode control. *IEEE transactions on control systems technology*, *7*(3), 328-342. IEEE. DOI: 10.1109/87.761053

[31] Ahmed, R., El Sayed, M., Gadsden, S. A., Tjong, J., & Habibi, S. (2016). Artificial neural network training utilizing the smooth variable structure filter estimation strategy. *Neural computing and applications*, *27*(3), 537-548. https://doi.org/10.1007/s00521-015-1875-2

[32] Mercorelli, P. (2014). An adaptive and optimized switching observer for sensorless control of an electromagnetic valve actuator in camless internal combustion engines. *asian journal of control*, *16*(4), 959-973. https://doi.org/10.1002/asjc.772

[33] Jahangiri, M., & Razaghian, F. (2014). Fault detection in analogue circuits using hybrid evolutionary algorithm and neural network. *Analog integrated circuits and signal processing*, *80*(3), 551-556. https://doi.org/10.1007/s10470-014-0352-7