

NEURAL NETWORK CONTROLLER FOR UPS APPLICATIONS

Petre-Daniel IONESCU¹, Alin MOSCALU², Mihaela MOSCALU³

¹⁾*Petre Daniel Ionescu, S.C. FORTUS S.A. Iasi*

²⁾*Alin Moscalu, RAJAC Iasi*

³⁾*Mihaela Moscalu, University of Medicine and Pharmacy "Gr.T.Popa" Iasi*

Abstract. This paper refers to an analogical controller with neural network for UPS invertors. Results of simulations performed for the proposed neural controller have shown that a low level for total harmonic distortions can be obtained, in conditions brought by non-linear loads. A comparison with an optimized PI controller has also been performed.

Results of comparison confirmed the superior performances of the neural network controller.

Keywords: neural network controller,

Introduction

UPSs are safety power sources, largely used for backing critical equipments, such as computers, automated process controllers or medical devices. Power converters of high efficiency are more and more used and, therefore, more and more non-linear electric loads appear, and more and more harmonics are generated. Hence, there is a great challenge for UPSs to maintain a quality sinusoidal output voltage, under non-linear loads [2], [3].

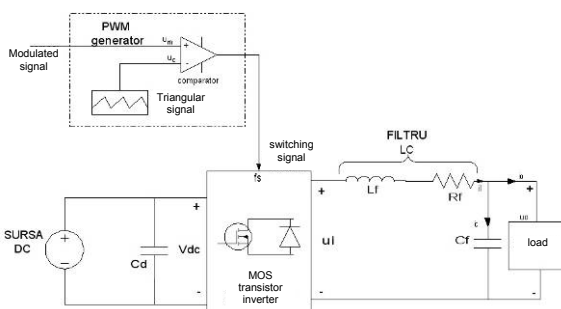


Figure 1. UPS inverter.

The block diagram of a typical UPS inverter is given in Figure 1. In order to have an optimal dynamic answer and low total harmonic distortions (THD) it is possible to adopt multiple loop control systems. Such a configuration works properly in the case of linear systems. But, if loads are non-linear, performances are poor. Recently there have been developed digital answer control circuits for PWM invertors. Despite that performances of these configurations are good; their implementing is difficult, due to complex algorithms and complex requests. In the last years,

neural networks have been adopted in many applications and been successfully implemented within power electronic circuits. As regards the control of DC/AC invertors, neural networks have been used for adjusting invertors' current for motors' control. In this case the neural network, depending on the phase current's error, generates a PWM signal which controls the inverter's switches [4] - [6].

Neural networks used within control systems can be on-line driven or off-line driven. In case of on-line drive, because neural network's weights and polarizations are adaptively modified during control process, in conditions of non-linear functioning, a better versatility is obtained.

The most popular driving algorithm for a neural network with positive reaction is the idle propagation. This algorithm is attractive due to its stability, robustness and efficiency. But it supposes a great number of multiplications and derivings [1]. If software is to be implemented, a very fast processor is required. If hardware is to be implemented, a very complex circuitry will result. Possible alternative for the idle propagation are algorithms based on perturbations, as the weights perturbation, the key-rules perturbation or the random modification of weights.

All these algorithms are using the perturbation of weights and gradients from generated errors are evaluated (instead of computing derivatives). These algorithms can be performed within VLSI analogical implementing [7]-[9]. Despite that,

for a real-time control of UPS inverters, some undesired outputs in network may occur, because the load conditions cannot be forecasted. There is possible to simulate a neural network in order to identify inverter's behavior with the goal to find neural network controller's output errors. But the simulated neural network has to be also pre-driven via data gathered from simulations or experiments.

An off-line drive of neural network requests a great number of pattern examples. These patterns can be obtained by means of simulations. Despite that weights and polarizations are fixed during the control process, the neural network is a non-linear system, much more robust compared to a linear system.

Moreover, neural network's direct computing mean only summations, multiplications and the correction of wave shapes with sigmoidal functions, which can be implemented with simple and cheap analogical hardware components.

For UPS inverters applications the optimal drive is the off-line drive for neural networks, which provides a fast response and can be cheaply implemented [10]-[12].

This paper proposes an analog neural network controller for UPS inverters which is to be off-line driven. Patterns are to be generated by controller simulations for an ideal reference current. In order to model this controller a feed forward neural network has been driven, using the idle propagation algorithm. After driving, the neural network is to be used for controlling the on-line inverter. Results of simulations show that proposed neural network controller leads to low total harmonic distortions, at non-linear load conditions, and a good dynamic response, in conditions of transitory regime. It is also described the implementation with programmable integrated circuits of the proposed analog neural network controller.

Proposed neural network controller

Figure 2 show the inverter's linear model, where the PWM inverter is modeled by a proportional block, with a gain $K = \frac{V_{dc}}{V_c}$ (where V_{dc} is the

voltage of DC power source, and V_c is the peak voltage of u_c triangular carrier presented in Figure 1).

A control diagram with multiple loops may provide good performances in conditions of known linear loads. Such a diagram measures the current within the capacitor (or inductor) of LC filter and uses the measured signal for controlling an internal loop. A control loop for external voltage is incorporated in order to obtain a sinusoidal output voltage. Due to the adoption of the internal control loop the inverter's output impedance is decreased and the dynamic rigidity gets amplified. Still, it is impossible to obtain a uniform performance in conditions of extreme loading, and the output voltage's distortion is significantly increasing when non-linear loads are used.

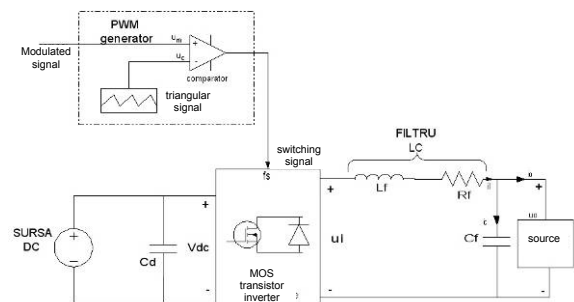


Figure 2. Linear model for the UPS inverter.

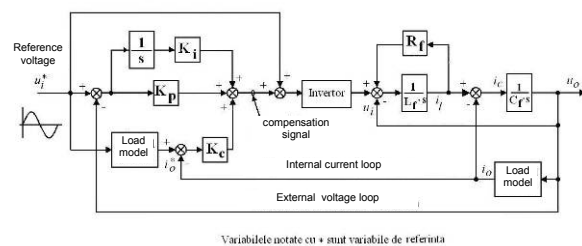


Figure 3. Controller with ideal load reference current, i_0^* , used to obtain pattern examples.

We are proposing a neural network controller for UPS inverters, in order to decrease output voltage's distortions for non-linear load conditions. An off-line driving is to be adopted, in order to ensure a fast transitory regime and low costs for this inverter. In order to produce good pattern examples for the driving of off-line neural network, we need to simulate a model which has to properly operate not only in

conditions of linear loading, but also in conditions of non-linear loading. The problem which occurs for non-linear loads is the high peak currents which distort the output voltage. If load current could be forecasted it would then be possible to design a controller in which the output current would be able to follow the prescribed current. Starting from a multiple control loop diagram, we will convert the internal current control loop on the capacitor (or inductor) to a control loop for current load, as it can be seen in Figure 3. A reference sinusoidal voltage feeds the model's load in order to generate an ideal reference current, i_0^* . The difference between the reference current, i_0^* , and the actual load current, i_0 , is used as input for the controller. The external voltage loop gets involved for generating a rectified output voltage. This model is easy to build and simulate. Its performances are good, not only for linear loads, but also in conditions of non-linear load. By means of MATLAB software such a controller has been simulated, for an ideal reference current. Within MATLAB the PWM inverter is described by the following equations:

$$u_i = \begin{cases} V_{dc} & (u_m \geq u_c) \\ -V_{dc} & (u_m < u_c) \end{cases} \quad (1)$$

where V_{dc} is DC source's voltage, u_m is the instant voltage of modulated signal and u_c is the instant voltage of PWM's triangular carrier (Figure 1).

The load model in Figure 3 can be of any type. In MATLAB resistors, inductors and capacitors can be easily built. A non-linear load, as a diode rectifying bridge, can also be built in MATLAB. A diode is described as it follows:

$$i_d = \begin{cases} 0 & u_d < 0,7 \\ (u_d - 0,7)/0,1 & u_d \geq 0,7 \end{cases} \quad (2)$$

where i_d is the instant direct current via the diode and u_d is the direct instant voltage on the diode.

It can be seen that a fixed set of parameters for the controller (K_p , K_i and K_0) is not suitable for all types of loads. Every type of load has a set of optimal parameters which can be computed by means of simulations, which generate a low THD output voltage and enough low stationary regime errors. The output voltage, the load

current and the current on inverter's capacitor are inputs of the neural network. The compensation (marked in the middle of Figure 3), instead to represent a global modulating signal, is in fact the desired output of the neural network. This helps to obtain a better learning and a better control of performances.

The neural network's off-line driving needs a large data base, which has to include the input-output relations. In case of UPS inverter the data base has to comprise the input-output patterns for all types of loads. Loads have been classified in two types: linear and non-linear. Afterwards, linear loads are divided in resistive, capacitive and inductive loads. Non-linear loads can be full rectifying bridges, followed by a filtering capacitor and a load resistor, and can be classified depending on their actual output powers, the power factors and peak currents. Every changing of the load type leads to a new set of patterns.

Data base comprises hundreds of patterns, of which two thirds are for linear loads, and the remainders of one third are for non-linear loads. Concerning the selection process of a neural network for the inverter, we think that this one has to be the simplest possible (few inputs and few hidden nodes) in order to keep the control process' speed high, and have low costs. The driving of the neural network is performed automatically, by means of computer software, which is randomly and repeatedly presenting to the neural network pattern examples selected from the pattern data base.

Each time, the neural network's weights and polarizations are updated, using the idle propagation algorithm, so that the square average error between the neural network's desired output and actual output to be lesser than pre-defined values.

Next, we are presenting an abstract of design stages for the neural network controller for UPS inverters applications.

1. Controller's simulation for an ideal reference load current, according to Figure 3.

2. For each type of load the controller's parameters are tuned to optimal values. As inputs for the neural network are used the output voltage, the load current and the capacitor's

current. The compensation signal will be the output of the neural network.

These patterns are in fact the pattern data base for the driving of the neural network.

3. A neural network structure is selected, in order to have a simple structure and also able for modeling the simulated controller, on basis of patterns stored in the data base.

4. The software neural network is driven (e. g. de MATLAB with Neural Network Toolbox).

Simulation's results and Hardware implementing

In order to check all items above described a neural network controller for UPS inverters has been designed. Its parameters are given in Table 1. In order to obtain the driving data a controller has been simulated, as the one shown in Figure 3, in MATLAB.

Table 1. Inverter's parameters

Parameter	Value	unit
switching frequency, f_s	20	kHz
source voltage DC, V_{dc}	48	V
nominal voltage	25	V_{rms}
nominal frequency	50	Hz
nominal current	5	A_{rms}
nominal impedance	5	Ω
filtering inductance, L_f	250	μH
coil's resistance, R_f	0,2	Ω
filtering capacitor, C_f	30	μF

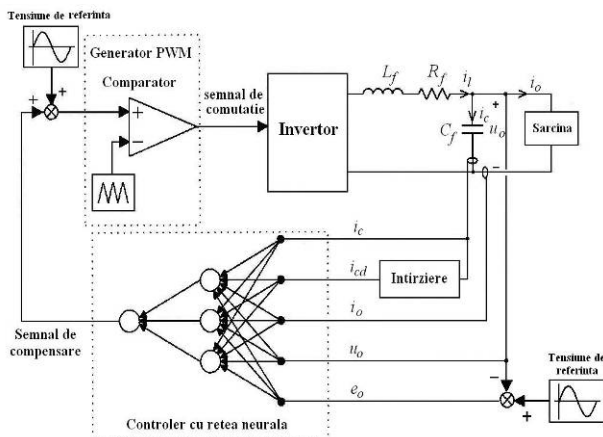


Figure 4. Diagram of a neural network controller proposed for controlling an UPS inverter.

Hundreds of patterns have been created for the pattern data base, for linear and non-linear loads. Figure 4 shows the neural network inverter's full diagram, in the 5-3-1 configuration (5 inputs, 3 nodes, 1 output). Hidden nodes have a sigmoidal transfer function. The output is a linear transfer function. This neural network structure resulted after several repeated attempts. Final structure is simple, but efficient. Its inputs are the current via capacitor, the delayed current via capacitor, the load current, the output voltage and the difference between a reference voltage and the output voltage. The delay period for the current via capacitor equals a switching period. Such a delay time is obtained with a simple low-pass RC filter. The neural network's driving is performed with MATLAB's Neural Network Toolbox.

A. Simulation's results

The proposed neural network controller will be simulated with PSpice, which uses precision models for diodes and switching components. Weights and polarizations within MATLAB simulations are integrated in the PSpice model.

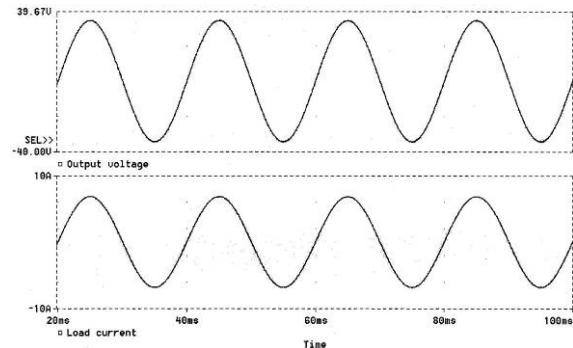


Figure 5. Results of simulating a permanent regime for the proposed neural network controller, for a full resistive load (5Ω).

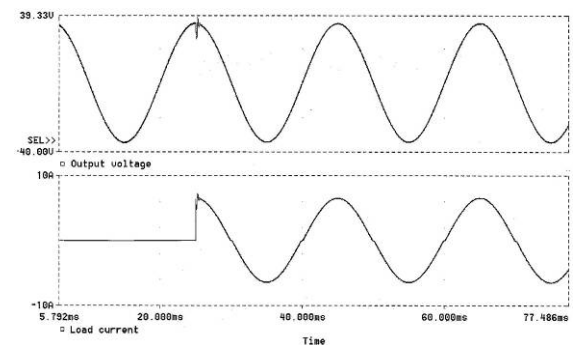


Figure 6. Results of simulation of a transitory regime for the proposed neural network controller, when load varies from load null to load maximum.

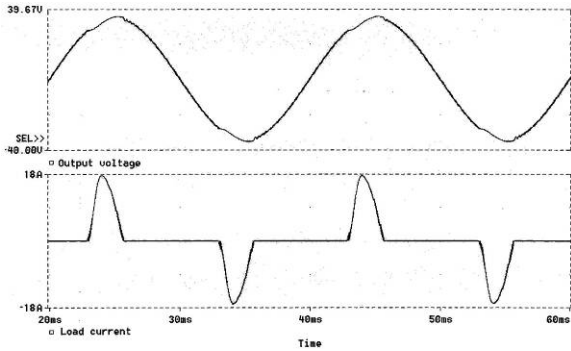


Figure 7. Results of simulation of proposed neural network controller, for non-linear load.

Will be investigated the response for transitory and permanent regimes for the proposed neural network controller. The output voltage's and invertors load current's wave shapes, for a fully resistive load (5Ω) are outlined in Figure 5. Figure shows that neural network controller is able to generate a proper sinusoidal output voltage. Figure 6 shows the results of simulating the dynamic response, when a load varies from zero to maximum (5Ω). Figure makes proof that system provides a fast dynamic response, with a slight altering of output voltage, this meaning that proposed controller is able to maintain a "compact" output voltage.

Figure 7 shows the results of simulation of proposed neural network controller, for non-linear load. That is, a rectifying diode bridge, followed by a $3200\ \mu\text{F}$ capacitor in parallel with a 10Ω resistor. It can be noticed that, despite the fact that load current has significant peaks, the voltage wave shape is just slightly distorted. THD for the output voltage is given by equation:

$$\%THD = 100x \sqrt{\sum_{n \neq 1} \left(\frac{\text{charmonic component}}{\text{fundamental component}} \right)^2} \quad (3)$$

B. The implementation of analogical hardware

After the off-line driving is completed, weights and polarizations do not change anymore during the control process. Because the direct propagation of neural network implies only multiplications, summing and the correction of wave shapes with sigmoidal functions, the implementing of proposed neural network is

feasible with a simple analog circuit. Such a simple analog circuit has been designed in order to implement this controller. Figure 8 shows the circuit diagram which mainly comprises operational amplifiers and resistors. The non-linear sigmoidal function is achieved by means of a differential pair. Neural network's weights and polarizations are represented by resistors. The first hidden node's output voltage, u_{nod1} , can be described as it follows:

$$u_{nod1} = \text{Sigmoid} \left[- \left(\frac{R_f}{R_1} u_{in1} + \frac{R_f}{R_2} u_{in2} + \frac{R_f}{R_3} u_{in3} + \dots + \frac{R_f}{R_b} u_b \right) \right] \quad (4)$$

where $\text{Sigmoid}(x) = \left(\frac{2}{1 + e^{-2x}} \right) - 1$.

It has to be noticed that analogical circuits feature a limited accuracy. The greatest deviation between the analogical implementation and the neural network is brought by the lesser accurate representation of resistor weights.

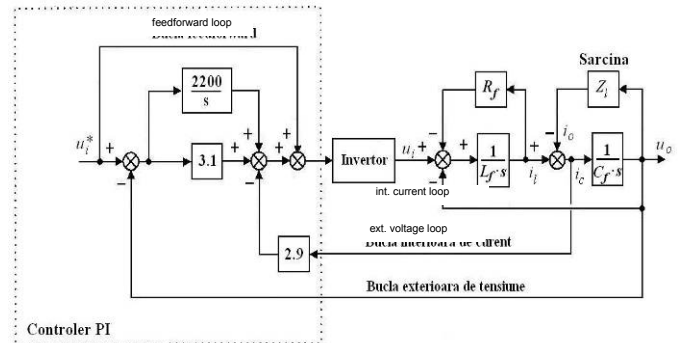


Figure 9. Analogical PI controller.

In order to compare we will study the functioning of an inverter controlled by a PI controller. The PI controller configuration is shown in Figure 9. Parameters of this controller have been optimized for fully resistive load conditions. It can be seen that the proposed neural network controller's output voltage has lesser distortions compared to the PI controller's output voltage.

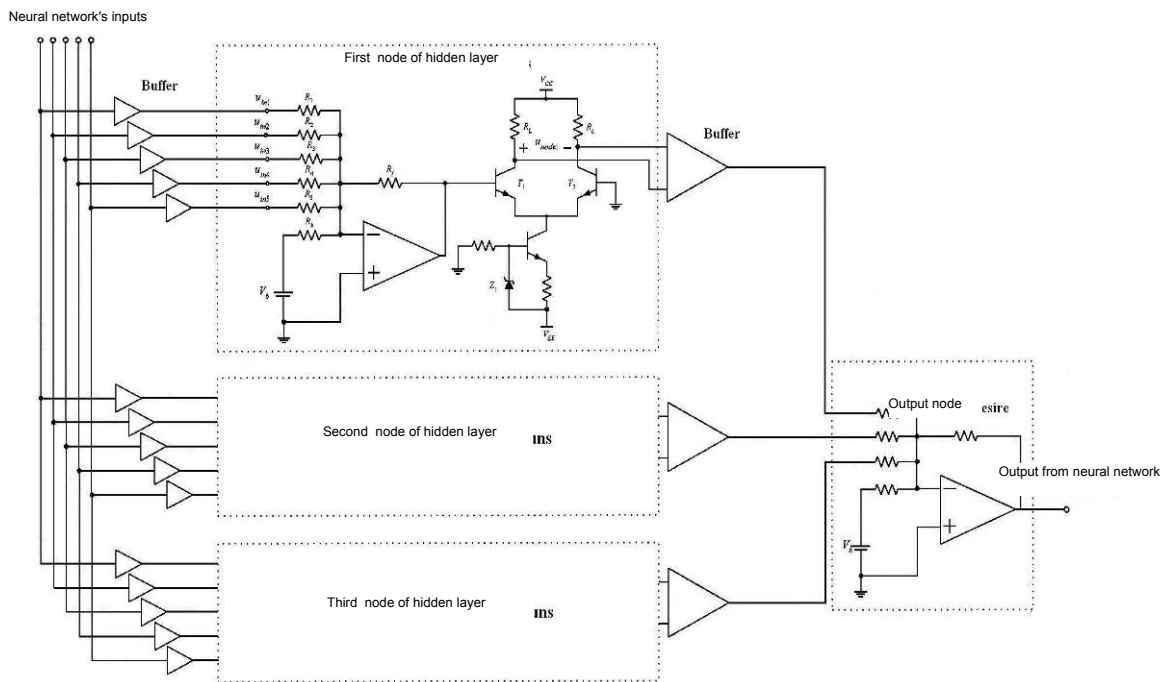


Figure 8. Analogical implementation of proposed neural network controller.

Figure 10 provides a comparison of THD for the output voltage in a neural network controller with the one in a PI controller, for various types of loads. It can be noticed that the proposed neural network controller's performances are significantly superior to the PI controller's performances.

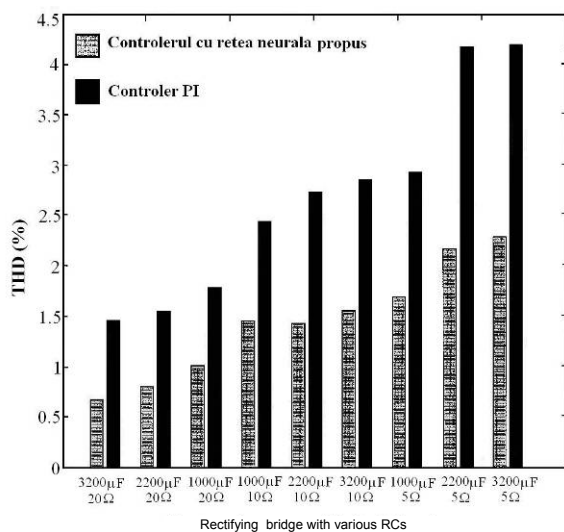


Figure 10. Comparison of output voltage's THD, for various types of loads.

For proposed neural network controller, the input-output map is stored in the pattern data base, which includes many cases of non-linear loads.

The neural network will "learn" the control rules from this data base, therefore the neural network will take into account the characteristics of these types of loads. This is the reason for which a neural controller generates lesser distortions on output voltage compared to PI controller. The neural network controller can easily be built with analogical circuits. Such a controller is able to maintain a permanent and transitory regime response, significantly decreasing the total harmonic distortions in conditions of non-linear load. Hence it is very suitable for applications where load brings periodic distortions.

Implementing of proposed controller with programmable integrated circuits

For proposed neural network controller described in section 3, if network's weights and polarizations have to be altered, the corresponding resistors are to be replaced. This is an element that hampers the manufacturing of

control circuits. Next we are proposing a combined circuit, analogical/digital, which can be implemented with digital circuits integrate, in order to model weights and polarizations. Figure 11 shows the block diagram of proposed integrated circuits, which are comprising three parts: the EPROM (for weights' storage), the synapses and the neurons. Following, there is a brief description of functioning for blocks within circuit proposed in Figure 11.

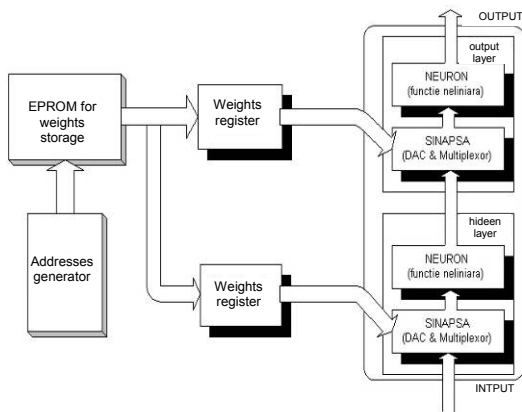


Figure 11. Block Diagram of integrated circuits within proposed neural controller.

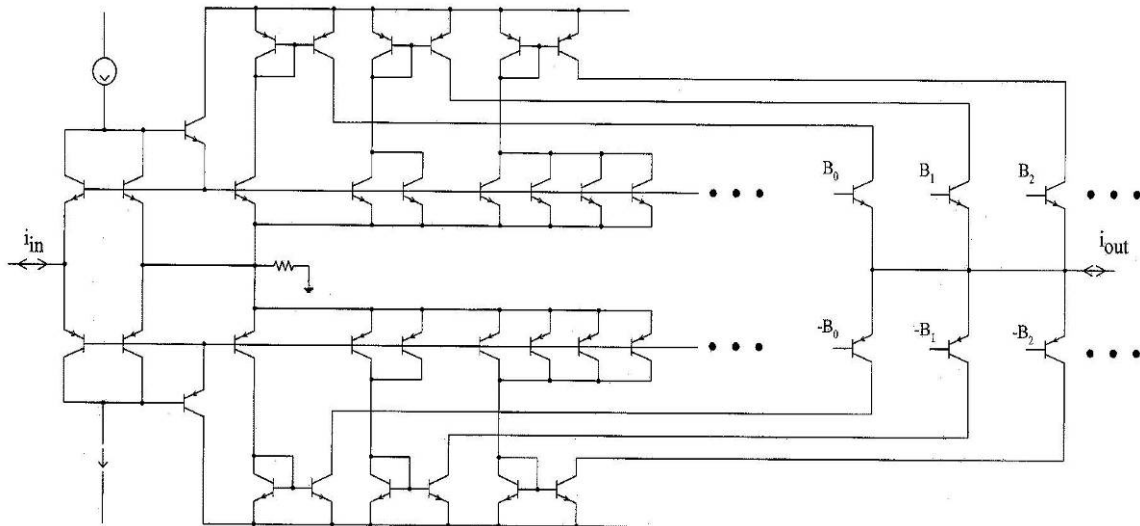


Figure 12. Circuit of a synapse.

The proposed programmable controller comprises only resistors and transistors, but can also be manufactured with integrated circuits. Such a configuration means fast computing and low costs, items that could fulfill the requests for real-time controllers for the power inverter.

A. The *EPROM* – Serves to digitally store the weights' and polarizations' values. Stored values in EPROM can easily be programmed.

B. *Synapses* – Receive weights' digital signal from EPROM and use it for programming the input weights' coefficient. In order to achieve this function we recommend the circuit shown in Figure 12. Input current, i_{IN} , which may come from the neural network's inputs or from the output of the last node, is used as reference for the current mirror. The output current, i_{out} , is the product of the input current and stored weight. This circuit works as a D/A converter and multiplexer.

C. *Neurons* – A differential pair which serves to model the sigmoidal functions, similarly to what it is shown in Figure 8.

Conclusions

This paper proposed an analog neural network controller for UPS inverters. Driving patterns for this controller are determined by simulating the controller under an ideal reference current. After driving, the neural network can be used for

controlling the on-line UPS inverter. Results of simulations show that proposed neural network controller can generate a low THD in conditions of non-linear loads and provides also a very good dynamic response during transitory regimes.

Proposed neural network controllers specifically suitable for non-linear loads which periodically are inducing distortions. There was also a discussion about the implementing of proposed controller with programmable integrated circuits.

Bibliography

- [1] Fodor Dan (1997) *Inteligență artificială în convertoarele statice de putere*, Curs, pp.28-59, pp.75-118,
- [2] N. M. Abdel-Rahim and J. E. Quaicoe, (1996) *Analysis and design of a multiple feedback loop control strategy for single-phase voltage-source UPS inverters*, IEEE Trans. Power Electron., vol. 11, pp. 532–541, July.
- [3] Ryan, M. J., Brumsickle, W. E. and Lorenz, R. D. (1997) *Control topology options for single-phase UPS inverter*, IEEE Trans. Ind. Applicat., vol. 33, pp. 493–501, Mar./Apr..
- [4] Kim, B. J. Choi, J. H., Kim, J. S. and Choi, C. H. (1996) *Digital control scheme of UPS inverter to improve the dynamic response* in Proc. IEEE Elect. Comput. Eng. Can. Conf., , pp. 318–321.
- [5] Jouanne, A. V. Enjeti, P. N. and D. Lucas, (1996) *DSP control of high-power UPS systems*

feeding nonlinear loads, IEEE Trans. Ind. Applicat., vol. 43, pp. 121–125.

[6] Tzou, Y. Y., Ou, R. S., Jung, S. L. and Chang, M. Y. (1997), *High-performance programmable ac power source with low harmonic distortion using DSP-based repetitive control technique*, IEEE Trans. Power Electron., vol. 12, pp. 715–725, July.

[7] Hunt, K. J., Sbarbaro, D., Zbikowski, R. and P. J. Gawthrop, (1992) *Neural networks for control system—A survey*, Automatica, vol. 28, no. 6, pp. 1083–1112,.

[8] Bose, B. K. (1994) *Expert system, fuzzy logic, and neural network applications in power electronics and motion control*, Proc. IEEE, vol. 82, pp. 1303–1323.

[9] Buhl, M. R. and Lorenz, R. D. (1991) *Design and implementation of neural networks for digital current regulation of inverter drives*, in Proc. IEEE-IAS Annu. Meeting, pp. 415–423.

[10] B. R. Lin and R. G. Hoft, (1993) *Power electronics inverter control with neural networks*, in Proc. IEEE-APEC, pp. 128–134.

[11] M. Trzynadlowski and S. Legowski, (1994) *Application of neural networks to the optimal control of three-phase voltage-controlled inverters*, IEEE Trans. Power Electron., vol. 9, pp. 397–404.

[12] D. Daniolos, M. K. Darwish, and P. Mehta, (1995) *Optimized PWM inverter control using artificial neural networks*, Electron. Lett., vol. 31, no. 20, pp. 1739–1740.